

SPECTRUM PARLANCHIN

WAFADRIVE: Análisis periféricos

EL ORDENADOR COMO PROYECTO PEDAGOGICO



¡¡MENUDO CAMBIO!!

Tráenos tu



SPECTRUM

y llévate un



SPECTRUM PLUS

Renuévate con INVESTRONICA.

Ahora INVESTRONICA te da la oportunidad de hacerte con el microordenador más moderno del mercado: EL SPECTRUM PLUS.
Sólo tendrás que entregarnos tu ZX SPECTRUM...
...lo demás será visto y no visto, el Spectrum Plus ya es tuyo.
Tener un ordenador Sinclair es la garantía de estar siempre a la última.

Apúntate a lo más nuevo.

El Spectrum Plus es lo más nuevo del mercado. Si tu Spectrum es estupendo; el Plus es fabuloso. Podrás disfrutar de un teclado profesional;
17 teclas más que el Spectrum, es decir 17 ventajas más... y por supuesto lo podrás utilizar con todos los programas y periféricos que ya tienes, puesto que **el SPECTRUM PLUS es totalmente compatible con todo el software y accesorios del spectrum.** Además INVESTRONICA, al realizar el cambio, **te da de nuevo 6 meses de garantía,** una nueva cassette de demostración y un libro de instrucciones a todo color.

No te lo pienses... cámbiate a lo último, tienes las de ganar.

Tenerlo, muy fácil

Manda tu ZX Spectrum (sin cables, ni fuente de alimentación) a tu Servicio Técnico Oficial (HISSA) más cercano, bien personalmente o por agencia de transportes (los gastos son por cuenta de INVESTRONICA) y en 48 horas ya podrás disfrutar de tu nuevo Spectrum Plus.
Sólo tienes que abonar (contra reembolso) 12.000 Pts. (*)



J M PUBLICIDAD

(*) 18.000 pts. si es de 16 K.

Dirígete a cualquiera de las delegaciones HISSA

C/. Aribau, n.° 80, Piso 5.° 1.°
Telfs. (93) 323 41 65 - 323 44 04
08036 BARCELONA

P.° de Ronda, n.° 82, 1.° E
Telf. (958) 26 15 94
18006 GRANADA

C/. San Sotero, n.° 3
Telfs. 754 31 97 - 754 32 34
28037 MADRID

C/. Avda. de la Libertad, n.° 6
bloque 1.° Entl. izq. D.
Telf. (968) 23 18 34
30009 MURCIA

C/. 19 de Julio, n.° 10 - 2.° local 3
Telf. (985) 21 88 95
33002 OVIEDO

C/. Hermanos del Rio Rodriguez, n.° 7 bis
Tel. (954) 36 17 08
41009 SEVILLA

C/. Universidad n.° 4 - 2.° 1.°
Telf. (96) 352 48 82
46002 VALENCIA

C/. Travesia de Vigo, n.° 32, 1.°
Telf. (986) 37 78 87
6 VIGO

Avda. de Gasteiz, n.° 19 A - 1.° D
Telf. (945) 22 52 05
01008 VITORIA

C/. Atares, n.° 4 - 5.° D
Telf. (976) 22 47 09
50003 ZARAGOZA

TodoSpectrum

DIRECTOR:
Simeón Cruz
COORDINADOR EDITORIAL:
Emiliano Juárez
REDACCION:
Juan Arencibia, Fernando García, José C. Tomás, Gumersindo García, Luis M. Brugarolas, Ricardo García, Santiago Gala
DISEÑO: Ricardo Segura

Editado por
PUBLINFORMATICA, S. A.
Presidente: Fernando Bolín
Director Editorial: Norberto Gallego

Administración:
INFODIS, S. A.
Gerente de Circulación y ventas:
Luis Carrero
Producción:
Miguel Onieva
Director de Marketing:
Antonio González
Servicio al cliente:
Julia González. Tel. 733 79 69
Administración:
Miguel Atance y Antonio Torres
Jefe de Publicidad
María José Martín
Dirección y redacción:
Bravo Murillo, 377-5.° A. Tel. 733 74 13
Telex: 48877 OPZX e 28020 Madrid
Administración y Publicidad:
Bravo Murillo, 377-3 E. Tels. 733 96 62/96
Publicidad Madrid:
María Jose Martín
Publicidad Barcelona:
María del Carmen Ríos, Olga Martorell. Pelayo, 12.
Tel. (93) 318 02 89.
08001 Barcelona.
Depósito legal: M-29041-1984
Distribuye S.G.E.L.
Avda. Valdelaparra, s/n.
Alcobendas-Madrid.
Fotomecánica: Karmat, C/. Pantoja, 10. Madrid.
Fotocomposición: Artecomp.
Imprime: Héroes, C/ Torrelara, 8. Madrid.
Esta publicación es miembro de la Asociación de Revistas de Información ARI asociada a la Federación Internacional de Prensa Periódica, FIPP.

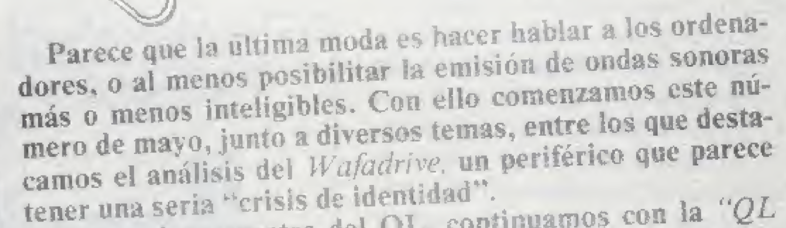
SUSCRIPCIONES:
Rogamos dirijan toda la correspondencia relacionada con suscripciones a:
TODOSPECTRUM
EDISA: Tel. 415 97 12
C/ López de Hoyos, 141-5.°
28002 MADRID
(Para todos los pagos reseñar solamente TODOSPECTRUM)
Para la compra de ejemplares atrasados diríjanse a la propia editorial
TODOSPECTRUM
C/ Bravo Murillo, 377-5.° A
Tel. 733 74 13-28020 MADRID

Si deseas colaborar en TODOSPECTRUM remite tus artículos o programas a Bravo Murillo 377, 5.° A. 28020 Madrid. Los programas deberán estar grabados en cassette y los artículos mecanografiados.
A efectos de remuneración, se analizará cada colaboración aisladamente, estudiando su complejidad y calidad.

Portada: LUIS GALA.

AÑO I • NUMERO 9





Parece que la última moda es hacer hablar a los ordenadores, o al menos posibilitar la emisión de ondas sonoras más o menos inteligibles. Con ello comenzamos este número de mayo, junto a diversos temas, entre los que destacamos el análisis del *Wafadrive*, un periférico que parece tener una seria "crisis de identidad".

Y para los amantes del QL, continuamos con la *"QL magazine"* en las páginas centrales. Un número reducido de páginas, para quienes quieren ir conociendo el nuevo ordenador Sinclair; pero suficientes, para aquéllos que no conocen ni desean conocer otro que no sea el Spectrum.

Hasta el próximo numero.

Spectrum

Algunos programas aparecidos en el mercado anuncian que son capaces de hacer hablar al Spectrum. Realmente se trata de un hecho curioso y divertido. A pesar de su aparente dificultad, es algo que pueden realizar en sus casas todos los usuarios de un Spectrum 48 K.

No se trata de sintetizar sonidos, ni de construir grandiosos artilugios que, enchufados al Spectrum, sean capaces de recitar la lista de reyes godos. Nuestra intención es almacenar en memoria un mensaje previamente grabado, para que el ordenador lo reproduzca posteriormente por el altavoz interno. Como más tarde explicaremos, el método no se caracteriza por una limpieza cristalina. En efecto, la señal viene acompañada por un fuerte ruido de fondo — hablando con propiedad diríamos que se obtiene un fuerte ruido de fondo al que se le suma la señal—. Sin embargo, el resultado es perfectamente inteligible, con una calidad semejante o superior a la de los programas comerciales que incorporan este tipo de efectos.

Principio de funcionamiento

Para grabar los sonidos tendremos que conectar el magnetófono, con un mensaje previamente grabado, a la entrada EAR, de forma semejante a cuando cargamos un programa.

La señal que entra en el ordenador es filtrada suavemente antes de aplicarse a un circuito digitalizador. Este proceso consiste en dar a la salida un nivel lógico 1 si la señal sobrepasa una cierta tensión o un cero si no se alcanza este valor. Para aclarar conceptos, observe cuidadosamente la figura 1. El "nivel de disparo" (que es el nombre que recibe) es constante e inalterable. Por ello es fácil de deducir que si variamos la amplitud de la entrada, la señal digitalizada asociada variará de una forma impredecible, siendo éste un resultado no deseable. De aquí, se deduce inmediatamente que el volumen con el que se reproduce la señal a grabar es algo crítico, y deberá ajustarse cuidadosamente.

Ahora estamos en condiciones de manejar esta señal, pues se trata de una señal digital, esto es, sólo distingue dos niveles de tensión. Se podrá, pues, almacenar en la memoria del ordenador. Sólo nos hace falta saber de qué manera.

Posponemos esta discusión para ocuparnos en primer lugar de la recuperación de la señal.

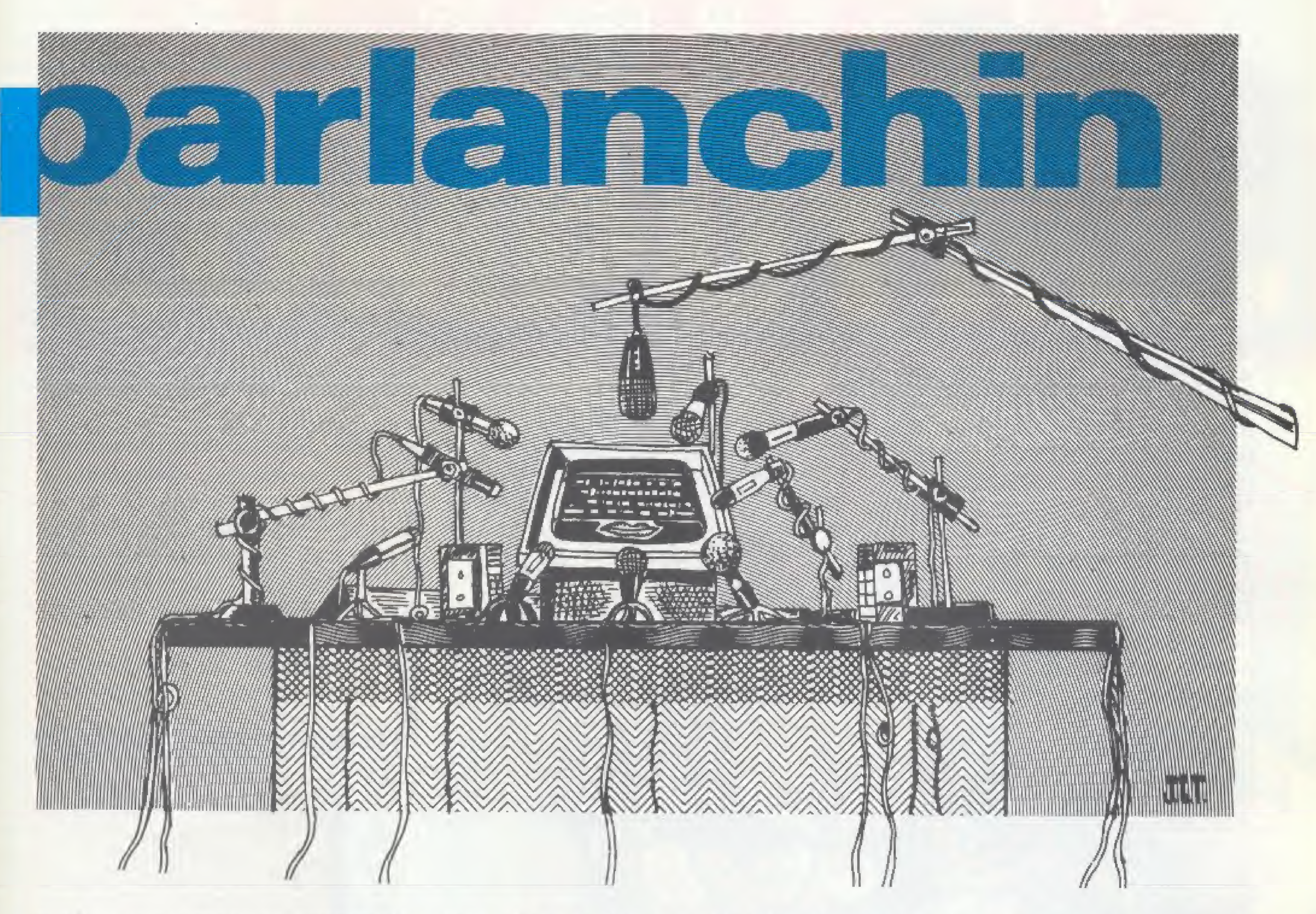
Recuperación de la señal

Teníamos una señal analógica "A", y procesándola hemos obtenido otra digital "D", que podemos almacenar de una manera (por ahora) totalmente mágica. Sin embargo, es claro que todo este proceso vale de poco si no somos capaces de recomponer la señal original o al menos una aproximación.

El sistema comúnmente empleado en la grabación o el tratamiento digital de señales analógicas requiere en primer lugar un muestreo periódico de la señal de entrada. Esto es necesario, ya que la señal digital tiene siempre una distribución discreta en la amplitud y en el tiempo.

La frecuencia de muestreo debe ser al menos dos veces más alta que el ancho de banda de la señal. Sólo con estas condiciones será posible recuperar la señal inicial (Teorema de Nyquist). Sobre las muestras se realizará una conversión analógico-digital, esto es, mediremos la tensión resultante con un número finito de cifras, lo que supone necesariamente un error. Para audio de alta calidad se usan 16 bits en la conversión (se miden 2^16 = 65536 niveles diferentes). Cuando se pretende conseguir una calidad aceptable, el usar menos de 8 bits es algo que ni se plantea. A la vista de esto, es lógico suponer que vamos a conseguir una calidad excepcional con un conversor de ¡un bit! Más de un sabio estudioso del tratamiento digital se sorprendería a la vista de los resultados obtenidos. Sin duda, calificaría el procedimiento de locura imposible; sin embargo, FUNCIONA.

Para reproducir la señal debemos dar salida a las muestras tomadas (que han sido almacenadas y tal vez procesadas de forma numérica) en el mismo orden en que han sido recogidas. Un filtrado paso bajo devuelve una señal tanto más parecida a la original, cuanto mayor sea el número de bits empleado en la conversión. En nuestro caso no vamos a hacer un filtrado serio, pero si lo hiciéramos obtendríamos una mejor relación señal-ruido. Al final obtenemos una señal bastante distorsionada a la



que hay que sumar un fuerte ruido ("ruido de cuantización"). El maravilloso cerebro humano se encarga del resto.

Almacenamiento del sonido

Vamos a volver al problema que teníamos pendiente. El método más intuitivo de almacenar la señal que vamos muestreando sería llenar la memoria de forma "lineal" con ceros y unos según sea el nivel lógico detectado a la entrada. Ver figura 2. El procedimiento es válido si la relación entre la frecuencia de la señal y la de muestreo es como se indica en la figura. Pero si aumentáramos la frecuencia de muestreo en cien o mil veces (o redujéramos la de la señal otras tantas), inmediatamente nos daríamos cuenta de que la densidad de información sería extremadamente baja. En efecto, supongamos que la frecuencia de la señal de la figura fuera 100 veces más baja. Esto significa que si en el primer intervalo de 1's contábamos tres muestras, ahora lo haremos 300. En el programa que mostramos, la frecuencia de muestreo es de unos 30 KHz. Usando 32 K de memoria sólo podríamos almacenar 8 segundos de "charla".

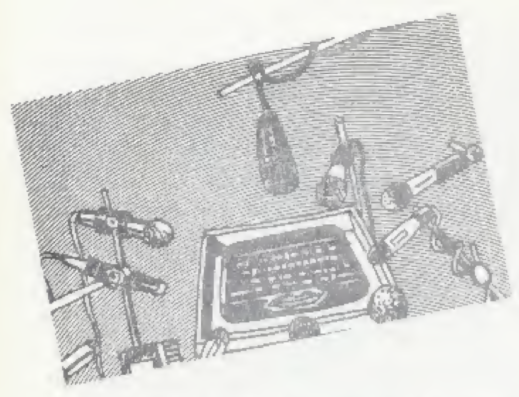
La solución es obvia: reduzcamos la frecuencia de muestreo, con lo que disminuiremos el ritmo de información requerido (número de bits por unidad de tiempo).

Sin embargo, lo que aparece como ingeniosa sugerencia presenta dos graves problemas:

— Si disminuimos excesivamente la frecuencia de muestreo, hasta cifras por debajo de la frecuencia de Nyquist (el doble del ancho de banda), nos encontramos con un fenómeno llamado **aliasing.** Este curioso efecto produce un plegamiento del espectro y genera un tipo de distorsión muy desagradable. Para comprender su funcionamiento, les remitimos a la figura 3. En el gráfico *a* vemos una señal que queremos muestrear. Se observa que parte de la señal tiene una frecuencia superior a la de muestreo. La figura *b* indica la señal que ha sido capaz de recoger el circuito muestreador. Podríamos decir que las componentes de alta frecuencia se le han "escapado". Y lo que es peor, la sección del espectro de las frecuencias más bajas han pasado a engrosar las filas de los tonos bajos.

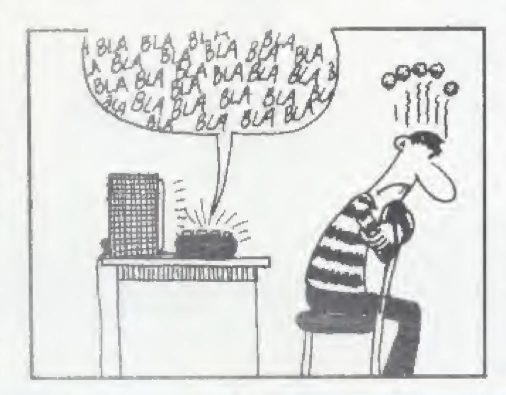
Si grabáramos un sonido cuya altura sube constantemente (como el sonido que hace un proyectil al caer) notaríamos cómo, a partir de un instante, la nota que se supone que debería seguir subiendo, se ha convertido en un tono grave. Esta nota alta tiene un "alias" (de aquí el nombre de **aliasing**). Resulta evidente que este efecto debe ser corregido. La forma evidente es realizar un filtrado previo, eliminando de esta manera las frecuencias altas que pudieran dar problemas. Y de paso disminuiríamos la densidad de información requerida. sin embargo, no hemos olvidado la promesa inicial de no usar cachivaches externos.

— Otro efecto pernicioso es que al bajar la frecuencia de muestreo de los 15-20 KHz, esta señal se



hace audible. De nuevo podríamos usar un filtro, también paso bajo, pero... Como usaremos el cassette para grabar en el Spectrum, es bastante razonable suponer que el ancho de banda es de unos 15 KHz. Por tanto usaremos una frecuencia de muestreo de aproximadamente 30 KHz. Ya hemos visto las consecuencias que tendría el bajarla. Hacerla más alta (aunque se pudiera), sólo llevaría a aumentar la memoria necesaria para grabar un cierto mensaje.

El método que hemos empleado consiste en contar el número de muestras que se toman entre dos transiciones de nivel. Así, si tenemos que la señal se ha mantenido en 1 durante 50 muestras y luego a 0 durante 30, tomaremos nota de estos datos, y la reproducción consistirá precisamente en poner a 1 la salida durante 50 períodos y después a 0 durante 30. Se observa



que, además de los tiempos, será necesario apuntar el estado lógico de la entrada. Una inversión lógica de los niveles (inversión de fase), no tendrá ninguna importancia siempre y cuando ésta sea constante.

Vamos a hacer algunos números par calcular ciertos aspectos de interés.

Con la frecuencia de muestreo escogida de 30 KHz, una señal muy asimétrica de 100 Hz necesitará 300 períodos de muestreo entre dos transiciones. Teniendo en cuenta que podemos encontrar fácilmente señales de frecuencia más baja, vemos que con un byte no tenemos bastante. Además está el problema de anotar el nivel de entrada.

Para permitir una gran flexibilidad, vamos a usar la siguiente con-

```
;        *************
;        ** COTORRA **
;        *************

;@ Luis Mig. BRUGAROLAS 1985

         ORG  8000H

;        GRABACION
;        ---------

RECORD  LD   HL,DATOS
        DI
        LD   C,01
LOOP2   LD   B,01111110B        ; 7
LOOP1   IN   A,(0FEH)           ; 11
        DEFB 0,0,0,0,0          ; 20
        DEFB 0,0,0,0,0          ; 20
        DEFB 0,0                ; 8
        RLCA ;                  ; 4
        RLCA ;                  ; 4
        AND  01                 ; 7
        XOR  C                  ; 4
        JR   NZ,FLANCO          ;70/125
        DJNZ LOOP1              ;80/135

        SLA  B                  ; 8
        LD   A,B                ; 4
        OR   C                  ; 4
        CPL  ;                  ; 4
        LD   (HL),A             ; 7
        DEFB 0,0,0,0,0          ; 20
        DEFB 0,0,0,0            ; 12

CONT    INC  HL                 ; 6
        LD   A,H                ; 4
        OR   L                  ; 4
        JR   NZ,LOOP2           ;125
        EI
        RET

FLANCO  SLA  B                  ; 8
        LD   A,B                ; 4
        OR   C                  ; 4
        CPL  ;                  ; 4
        LD   (HL),A             ; 7
        LD   A,C                ; 4
        CPL  ;                  ; 4
        AND  01                 ; 7
        LD   C,A                ; 4
        JR   CONT               ; 12


;        REPRODUCCION
;        ------------

PLAY    LD   HL,DATOS
        DI

LOOP4   XOR  A                  ; 4
        LD   B,(HL)             ; 7
        SRL  B                  ; 8
        JR   C,LOOP3            ;70/125
        LD   A,0FFH             ; 7

LOOP3   OUT  (0FEH),A           ; 11
        DEFB 0,0,0,0,0          ; 20
        DEFB 0,0,0,0,0          ; 20
        DEFB 0,0,0,0,0          ; 20
        DEFB 0,0,0,0            ; 16
        DJNZ LOOP3              ;80/135

        DEFB 0,0,0,0,0          ; 20
        DEFB 0,0                ; 8
        INC  HL                 ; 6
        LD   A,H                ; 4
        OR   L                  ; 4
        JR   NZ,LOOP4           ;125
        EI
        RET


DATOS   EQU  $
;       Aqui viene el 'discurso'
;       hasta el final de la me-
;       moria.
```

SPECTRUM COMPUTING

ESPECIAL AVENTURAS

DOS GRANDES JUEGOS EN CODIGO CON MAQUINA OPCION DE JOYSTICK

JUEGOS

Chopper PILOTANDO UN HELICOPTERO, TENDRA QUE DIRIGIR EL EQUIPO DE RESCATE PARA SALVAR A LOS ABANDONADOS EN UN CAMPO PETROLIFERO, DE UNA MUERTE SEGURA.

Convoy DESTRUIR LOS ALIENIGENAS Y SUS NAVES ES LA UNICA FORMA DE SALVAR LA TIERRA DE LA INVASION.

ARTICULOS

Twiddler MUESTRA LOS MISTERIOS DEL RAPIDO CAMBIO DE COLOR.

Cartoon APRENDA A PROGRAMAR DIBUJOS ANIMADOS.

Hangout CONOZCA MAS A FONDO LAS POSIBILIDADES DE SU ORDENADOR.

y mucho más

875 ptas.

BIENVENIDO A

SPECTRUM COMPUTING

CHOPPER
TWIDDLER
SHOOT
HANGOUT
TOMATOES
CARTOON
CONVEYOR
TALLER
CONVOY
LIGHTBIKE
LA TUMBA DE ELLAK

¡Ya está a la venta! Consúltela en su quiosco habitual

SORTEO ESPECIAL

MAS DE 150.000 PTAS. EN PREMIOS BASES EN EL INTERIOR

Solicítela a: **INFODIS** C/ Bravo Murillo, 377 - 5.º A - 28020 MADRID

CUPON DE PEDIDO · **N.º 3**

Sí, envíenme al precio de 875 ptas.... ejs de SPECTRUM COMPUTING

El importe lo abonaré: Contra reembolso ☐ Con mi tarjeta de crédito ☐

Adjunto cheque ☐ American Express ☐ Visa ☐ Interbank ☐

Número de mi tarjeta ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Fecha de caducidad ________

NOMBRE ________

DIRECCION ________

CIUDAD ________

PROVINCIA ________

Sin gastos de envío

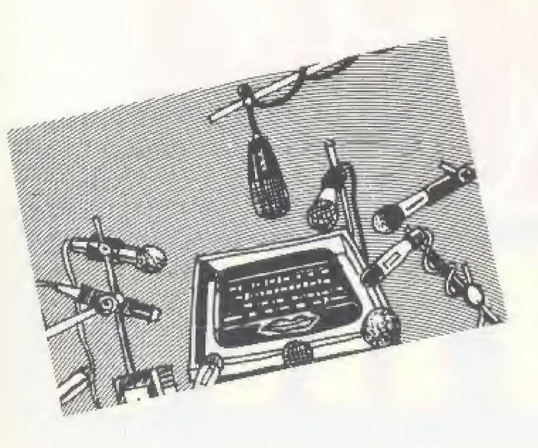


Figura 1

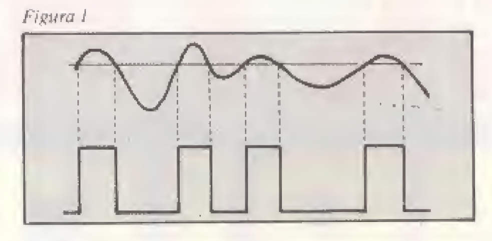
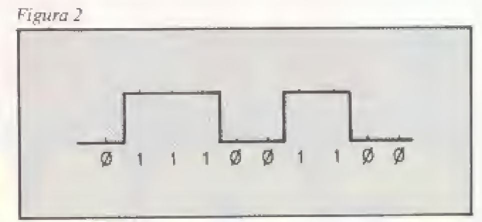


Figura 2



figuración: en cada palabra de 8 bits reservaremos uno (el menos significativo) para indicar el estado lógico que corresponde a la señal (si es 0 ó 1). Los siete bits restantes se usarán para almacenar el número de ciclos de muestreo que la señal ha estado en el nivel dado. Por tanto, el valor almacenado será $2 * n + e$, donde "n" es el número de muestras tomadas y "e" es el valor de la entrada.

Aclaremos las ideas con un ejemplo (ver fig. 5). En primer lugar, tendremos la entrada a 0 durante 5 ms. Con el valor dado de la frecuencia de muestreo, durante este tiempo habremos sido capaces de tomar $5.10^{-3} * 3.10^{4} = 150$ muestras. Como en cada byte podemos contar un máximo de $2^7-1 = 127$, necesitamos encadenar dos. El primero con el valor de 254 y el segundo con 46, el doble de los 32 restantes.

A continuación la entrada estará en 1 durante 100 us. Podremos en este tiempo muestrear la entrada $10^4 * 3.10^4 = 3$ veces. Almacenaremos por tanto el número 7.

Programas

Vamos a pasar directamente a la descripción de los programas. El primero de ellos es el encargado de la grabación (RECORD) y el segundo de la reproducción, PLAY.

El área de memoria asignada para el almacenamiento de los discursos es la comprendida entre la dirección START y el final de la memoria FFFFH. Es interesante que tanto el programa como el archivo de "voces y gritos" estén en el segundo bloque de 32 K, esto es, entre las direcciones 8000H y FFFFH. La justificación del hecho reside en que el acceso a memoria en el primer bloque de 16 K de RAM sufre pequeños retrasos debido al robo de ciclos que efectúa la ULA. Esto podría afectar a los procesos de temporización del programa.

Por idénticas razones, aunque en este caso mucho más justificadas, es necesario deshabilitar las interrupciones (mediante la instrucción DI, ver TODOSPECTRUM número 6).

Programa RECORD

Usaremos el registro C para almacenar información relativa al estado de la entrada. El contenido del bit menos significativo de C será un fiel testigo del nivel de entrada. Para detectar una posible transición de nivel compararemos este registro con el A, después de haber hecho sobre éste algunas manipulaciones. Veamos cuáles son éstas.

Al acabar de ejecutar una instrucción de entrada de datos a través del puerto 254 (FEH), nos encontramos en el bit 2 de A la información relativa a la entrada de *cassette* (quienes se hayan sentido ligeramente indispuestos, les recomendamos una cura a base de una lectura del capítulo 23 del manual).

Si realizamos dos giros sucesivos del acumulador hacia la derecha, el mencionado bit aparecerá en el lugar menos significativo. Al hacer un AND 01, despejaremos la palabra de información no deseada. El acumulador guarda ahora un 0 ó 1 en función de la entrada. Comparando este registro con el C, estaremos en condiciones de determinar si ha habido o no cambios de nivel.

El bucle de muestreo está controlado por un DJNZ. El agudo lector habrá observado que el re-

Programa 2

```
   1 REM ** PROGRAMA CARGADOR **
   2 REM
  10 DATA 33,115,128,243,14,1,6,
126,219,254,0,12
  20 DATA 7,7,230,1,169,32,24,16
,233,203,32,120,177,47,119,0,9
  30 DATA 35,124,181,32,211,251,
201,203,32,120,177,47,119,121,47
,230,1,79,24,236
  40 DATA 33,115,128,243,175,70,
203,56,56,2,62,255,211,254,0,19
  50 DATA 16,233,0,7
  60 DATA 35,124,181,32,213,251,
201
 100 RESTORE : CLEAR 30000
 110 LET A=32768: LET I=0: LET S
UMA=0
 120 IF I>114 THEN  GO TO 200
 130 READ D: IF D=0 THEN  READ F
: FOR J=1 TO F: POKE (I+A),0: LE
T I=I+1: NEXT J: GO TO 120
 140 POKE (I+A),D: LET SUMA=SUMA
+D: LET I=I+1: GO TO 120
 200 IF SUMA<>8176 THEN  GO TO 1
000
 210 PRINT " Programa correctame
nte escrito."''TAB 6;"Grabelo en
 cinta"'''': SAVE "Vox"CODE A,I:
 PRINT "     Rebobine para verifi
car": VERIFY ""CODE A,I: STOP
1000 BEEP .2,30: PRINT "Hay un e
rror. Revise los DATA's": STOP
```

Figura 3



Figura 4

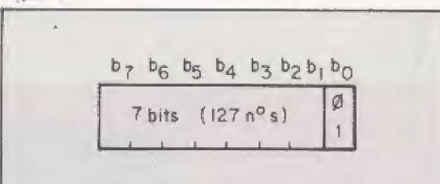
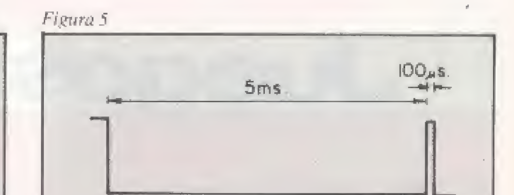


Figura 5



gistro B ha sido previamente inicializado con el máximo número de períodos que puede almacenarse en un byte (recordemos que un bit se reserva para apuntar el nivel). Según se van tomando muestras sin que se altere el nivel de entrada, se decrementa el registro B. Es fácil de demostrar que en B tenemos el complemento a 1 (la inversión lógica) del número de cuentas realizadas, a excepción del mencionado bit de nivel. En la práctica rotaremos B, introduciremos el ya famoso bit, complementaremos y escribiremos en memoria.

El bucle de muestreo se abandonará en dos posibles circunstancias:

1) Se realiza un ciclo completo de búsqueda sin detectar cambios de nivel. En este caso se realiza la operación descrita, se incrementa HL (dirección a introducir el próximo dato) y se comprueba si hemos llegado al final. Para esto haremos un OR entre H y L. Sólo en el caso de que ambos registros estén a 0 (esto es, HL = 0000) el resultado será cero, lo que se reflejará en el Zero Flag. Si llegamos al final devolvemos la posibilidad de responder a las interrupciones y pasamos al BASIC.

2) Se detecta un cambio a la entrada. En este caso acudiremos a FLANCO, donde se invierte el contenido de C y se repiten las operaciones del caso anterior.

Programa PLAY

Esta rutina es mucho más sencilla. En función del bit de nivel cargaremos en A todo ceros o todo unos. Al ejecutar la instrución OUT (FEH), A, el contenido del acumulador se "desparramará" por el exterior manchando el borde de diferentes colores, teniendo también eco en el altavoz exterior.

El proceso —muy semejante al anterior—, consistirá en repetir las salidas en la misma secuencia de entrada.

DEFBO, ¡qué raro!

Es realmente poco frecuente encontrar este tipo de cosas en programas razonables (¡lo que ocurre es que éste no lo es!). El DEFB *(DEFine Byte)* es una instrucción del lenguaje ensamblador del Z 80 que no tiene equivalente en el código máquina de este uP. Es lo que se llama "pseudo op-code" (op code: código de operación). Su misión es la de poner en memoria el byte indicado, en este caso 0. El cero corresponde a una instrucción de código máquina que todo el mundo conoce bien, el NOP, cuya misión es básicamente la de no hacer nada (esto es, perder el tiempo miserablemente).

No hay que asustarse de los DEFB ya que en nuestro caso resultan ser inofensivos NOP's. No se ha empleado esta notación más clara, ya que entonces el listado resultaría enormemente largo.

Ya sabemos qué son. Ahora sólo nos queda conocer para qué sirven. Hemos insistido mucho en que los procesos de grabación y reproducción han de llevarse a cabo con una puntualidad exquisita. De otra manera los resultados serán frustrantes. Quien disponga de un ensamblador y tiempo libre, puede comprobar resultados añadiendo o quitando ceros, esto es, alterando la temporización del proceso.

Lo que se pretende es, simplemente, igualar el tiempo que tardan en ejecutarse los diferentes bucles, sea cual sea el camino que se tome o la rutina en la que estemos. Estas "pérdidas de tiempo" controladas son una inestimable ayuda para conseguir nuestros propósitos.

En el listado en Ensamblador del programa se han indicado en la columna derecha el número de períodos de reloj necesarios para la ejecución de cada instrucción. Dejamos para el lector la comprobación de lo anteriormente expuesto. Recordamos que el Z 80 del Spectrum tiene una frecuencia de reloj de 3.5 MHz, lo que significa que un período dura 290 ns.

Confiamos que disfrute de unas agradables conversaciones con su amigo el Spectrum. ¡Y, en cualquier caso, siempre es fácil hacerle callar!

Luis M. Brugarolas

Programa 3

```
  10 CLEAR 30000: LOAD ""CODE
  20 LET REC=32768: LET PLAY=328
34
  30 PRINT " Ponga en marcha el
casette."''" Apriete alguna tecl
a para      empezar a grabar"
  40 FOR A=0 TO 100: NEXT A: PAU
SE 0: CLS : PRINT AT 10,13; FLAS
H 1;"GRABANDO": LET A=USR REC
  50 CLS : BORDER 0: PRINT "Apri
ete:E para escuhar
    G para grabar"
  60 INPUT A$
  70 IF A$="g" THEN  GO TO 30
  80 IF A$="e" THEN  LET A=USR P
LAY
  90 GO TO 50
```

CHESS THE TURK

Distribuidor: Sinclair Store.
Precio: 1.250 Ptas.

Dejando aparte la deficiente traducción, el único problema con el que nos encontramos es un juego "demasiado inteligente" por parte del ordenador: no nos deja hacer jugadas ilegales y cuando le pedimos ayuda nos orienta hacia jugadas "desastrosas". Un talento.

El menú da una orientación rápida de las posibilidades del juego, hasta 14 opciones diferentes. Veámoslas en forma resumida.

1. Juego nuevo. Para comenzar a jugar.
2. Continúe juego antiguo. Para seguir un juego anterior.
3. Ajedrez blitz. Partidas contrarreloj.
4. DEMO. El Turco juega contra sí mismo.
5. INPUT secuencia de pasos. Para dos jugadores.
6. Pasos retrazables en el juego. Visión "retrospectiva" de las jugadas.
7. EDIT pantalla o nueva posición. Para partir de una posición determinada.
8. LIST pasa a pantalla. Listado de movimientos en pantalla.
9. LIST pasa a imprenta. Listado de movimientos en impresora.
10. LINE PRINT la pantalla. Impresión de la pantalla.
11. SAVE pasa a cinta. Se graban todos los movimientos.
12. SAVE pantalla a cinta. Se graba la pantalla.
13. LOAD pasa de la cinta. Se cargan todos los movimientos.
14. LOAD pantalla de la cinta. Se carga la pantalla.

Variadas opciones para todos los gustos, con buena presentación. La opción 6 resulta especialmente vistosa al poder visualizar los movimientos de toda la partida, con desplazamiento pixel a pixel vertical de las casillas afectadas. La entrada de jugadas es otro buen ejemplo de la cuidada presentación: cuando se solicita ayuda se coloca la jugada en la línea de entrada y basta con teclear ENTER.

Todo esto está muy bien, pero además juega. Tiene seis niveles de juego, siendo los tiempos de respuesta aproximados de 5, 10 y 90 segundos para los tres primeros niveles; 10 y 60 minutos para el cuarto y quinto, y hasta 6 horas para el sexto y último nivel.

Mientras "piensa", se visualizan en pantalla las distintas posiciones que el ordenador evalúa, antes de tomar la decisión definitiva.

Pantalla de carga de "El Turco".

Opción 1 del menú, con los relojes en el margen derecho. Opción 6 del menú. El margen derecho muestra los distintos movimientos.

Control: *Teclado.*
Jugadores: *Uno o dos.*
Gráficos: *Se reconocen claramente las piezas, utilizándose tonos distintos según la opción inicial elegida de color o blanco y negro.*
Sonido: *No tiene.*
Niveles de dificultad: *Seis.*
Originalidad: *Clásico juego del ajedrez.*
Conclusión: *Ciertamente no es un juego que podamos calificar de original o novedoso. Lo verdaderamente importante de todo juego de ajedrez es que pueda desarrollar una capacidad de cálculo aceptable sin grandes demoras de tiempo. Este cumple este requisito y además ofrece un variado menú de opciones para disfrutar del juego/deporte del ajedrez en sus diversas acepciones.*

BRUCE LEE

Distribuidor: Erbe.
Precio: 2.100 ptas.

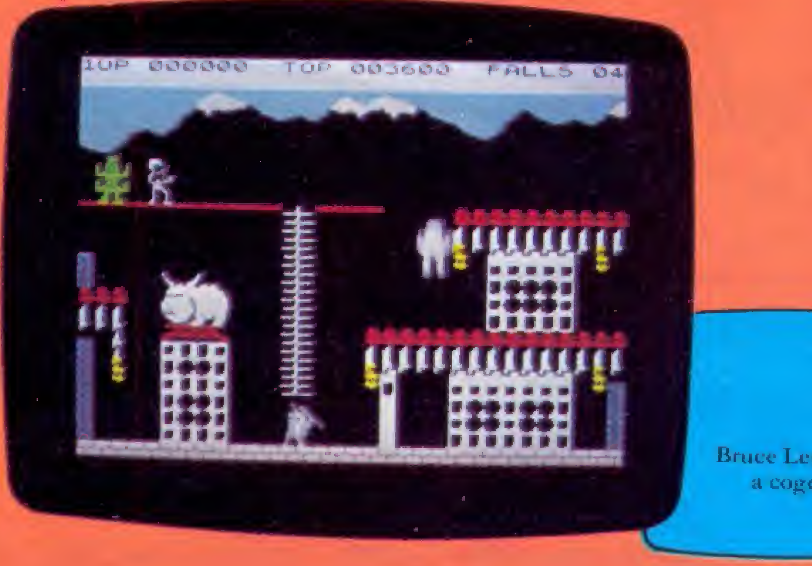
De inminente "estreno" para este mes, el juego de Bruce Lee tiene todos los requisitos para entrar en la lista de los más vendidos: una aventura en la que hay que sortear diversos peligros para llegar al objetivo final. Objetivo que en este caso es llegar a la morada del "Gran Sabio", para obtener su magia y poder.

No se trata de un juego contrarreloj, y esto es importante tenerlo en cuenta a la hora de entrar en pelea o ir hacia el objetivo final. La lucha, lógicamente, se realiza mediante las artes marciales, consistente en golpes con las manos o con los pies, para defenderse del enemigo que constantemente nos persigue. Los efectos de la lucha están tan bien conseguidos, que sólo con esto ya se justificaría el juego.

En cuanto a los "enemigos", sólo son dos, pero su efecto combinado puede ser mortal. Tres golpes suponen la muerte, disponiendo de cuatro vidas por juego. También ellos "abandonan la existencia" después de recibir tres golpes, pero tienen un número ilimitado de vidas, con lo que sólo nos libramos de ellos temporalmente.

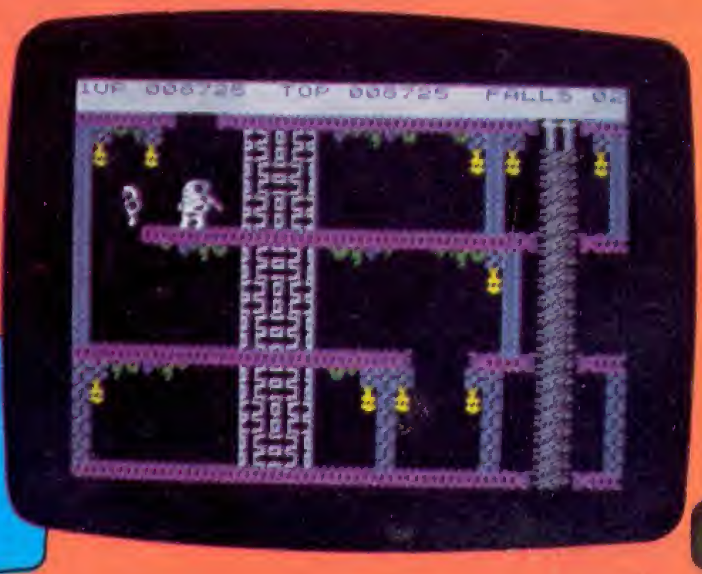
Pero si lo que se quiere es alcanzar el objetivo final, para ello no sólo habrá de salir victorioso de las contiendas con el enemigo. El paso por el "laberinto de túneles" se consigue mediante las linternas que hay que atrapar con diversos saltos (en ocasiones de un solo intento). A ello se añade una dificultad adicional: el factor aleatorio de determinadas plantas que se abren y devoran a todo lo que pillan.

Las distintas pantallas están bien conseguidas, con efectos gráficos de buen calidad, pero sin duda su mayor originalidad y adicción lo constituye la lucha "cuerpo a cuerpo" con los incansables karatekas. Ello unido a la necesidad de ir avanzando lentamente para alcanzar el objetivo final.

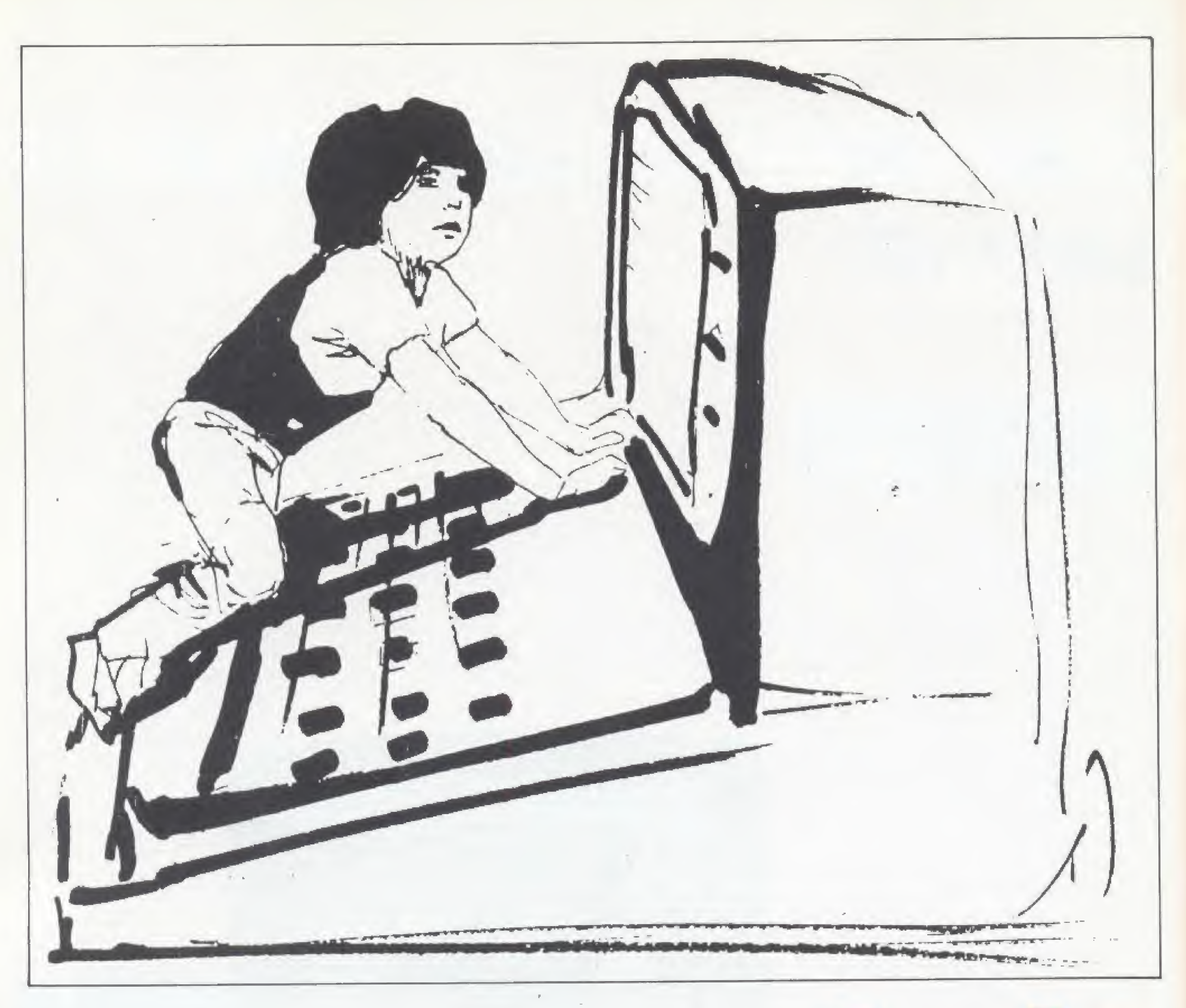


Bruce Lee se dispone a coger los faroles amarillos.

Control: *Teclado, Joystick.*
Jugadores: *Uno o dos contra el ordenador o entre sí.*
Gráficos: *Múltiples pantallas con buenos y variados gráficos.*
Sonido: *Distintos tonos que pueden activarse o desactivarse.*
Niveles de dificultad: *No hay niveles, aunque la dificultad va en aumento según se va avanzando.*
Originalidad: *Buen juego sin competencia en el mercado, por el momento.*
Conclusión: *La adicción es su mejor característica, ya sea en la utilización de las "técnicas marciales", en la búsqueda del "gran mago", o mejor aún, en combinar ambas situaciones.*



Bruce Lee pasa a la segunda fase del juego, en busca de "El Gran Mago".



Aula informática con Spectrum

y 2

Comentábamos en el número anterior la problemática que representa la introducción de la informática en centros escolares, tanto en educadores como para profesionales de la informática.

¿Qué podemos hacer con los ordenadores en la escuela? La respuesta a esta pregunta no es sencilla. La enseñanza tiene una compleja organización de niveles, cursos, programas de estudio y distintos propósitos según las edades del alumnado. Lo que podremos hacer en estos diferentes estamentos dependerá

mayormente de los planteamientos pedagógicos que los rigen y que muchas veces son insuficientemente claros.

Muchos sostienen que la educación primaria tiene como eje central la formación del alumno en habilidades básicas de lectura, escritura y conocimientos generales de ciencias y letras. Sin embargo, este acento en la **formación** está frecuentemente contrastado con un terrible bombardeo de **información.** Tremendos manuales de diferentes asignaturas, enciclopedias y diccionarios rodean al niño, contradiciendo lo que aparentemente debería ser. Parece que la habilidad más importante es la de memorizar información para más tarde escupirla en un examen. ¿Cuántas veces hemos oído las quejas de los profesores de matemáticas por las dificultades que tienen sus alumnos para resolver los dichosos "problemas"? Son estas habilidades las que quizá deberíamos desarrollar mucho más. La posibilidad de trabajar los aspectos más creativos del intelecto, de construir herramientas sólidas de razonamiento, de potenciar en última instancia el desarrollo intelectual de niños y jóvenes está fuertemente vinculada al uso que podemos hacer del ordenador. El LOGO es un lenguaje de programación diseñado especialmente con esta finalidad. Claro está que si lo que pensamos es que nuestros alumnos deban memorizar la Enciclopedia Británica como su mayor bien cultural, difícilmente podremos utilizar un ordenador que esté al alcance de nuestros medios económicos.

El uso de los ordenadores depende de los criterios pedagógicos generales que tengamos y de la labor que deseemos realizar.

> *Un profesor puede utilizar los ordenadores sin ningún conocimiento de programación*

El tipo de enseñanza determina en mucho lo que podemos hacer con un aula informática. No haremos lo mismo en una escuela de Básica que en un centro de formación profesional administrativa. Es quizá innecesario que los alumnos de BUP trabajen con una tabla gráfica, pero es muy útil en una escuela de diseño. Lo que sí podemos establecer es un criterio general que nos sirva para todas estas distintas enseñanzas.

Con un aula informática podemos desarrollar un trabajo en dos frentes: trabajar con lenguajes de programación y trabajar con programas educativos.

Entendemos por programas educativos no sólo aquellos que hayan sido concebidos para tratar un determinado tema, ya sea de Geografía o de Química, o para corregir en el mismo ciertas dificultades en la escritura. También pueden utilizarse ciertos programas como un procesador de textos con una finalidad educativa. Un procesador de textos, originariamente concebido para confeccionar cartas e informes en una empresa, puede convertirse en una hermosa herramienta en manos del profesor o la profesora de literatura. ¿Por qué no escribir la redacción con el ordenador? ¿No es posible acaso realizar un periódico estudiantil con este programa, creando así una actividad enriquecedora?



Sobre este particular, es decir, sobre el *software* educativo, nos centraremos en futuros artículos.

Este criterio general que hemos expuesto sobre el uso de un aula informática puede aclarar también un tema que preocupa a aquellos maestros y profesores que no se sienten muy entusiastas hacia la informática.

Con razón, muchos ven en esto

la necesidad de estudiar un tema nuevo y desconocido; la necesidad de reciclarse casi obligatoriamente en algo que no eligieron vocacionalmente.

Esta confusión quizá se deba a que se reduce la utilización del aula informática a la enseñanza de lenguajes de programación.

Para la secretaria de una empresa basta con que sepa cómo encender el equipo, cómo cargar el programa y conocer el funcionamiento del mismo para que pueda realizar sus anotaciones contables. No necesita en absoluto conocimiento alguno de programación y sin embargo puede realizar eficientemente su trabajo.

De igual forma, un profesor puede utilizar los ordenadores del centro sin ningún conocimiento de programación. Basta que sepa cómo funciona el aula, cómo se cargan los programas y cómo utilizarlos para sacarles un buen provecho.

Con esto no queremos decir en absoluto que deban limitarse a ello, como tampoco pensamos que el aprendizaje de ciertos lenguajes de programación no les sea de uti-





Plus y Logo: Dos buenos

El Colegio Zurich de Barcelona, dedicado a la Enseñanza General Básica, es un buen ejemplo de aula informática con Spectrum. José Macías, director del centro, nos relata sus actividades: "Actualmente tenemos doce Spectrum Plus, por los que pasan todos los alumnos. Nuestro interés es la adecuación de la informática a la estructura pedagógica general del centro, a partir de los departamentos de ciencias y más concretamente de matemáticas, donde vemos aplicaciones inmediatas. Pero también puede haber otros departamentos que se aprovechen de ello, utilizando los microordenadores con programas educativos.

No somos partidarios de la enseñanza asistida por ordenador, y para nosotros la alternativa son lenguajes como el Logo. Al niño le resulta muy difícil seguir un razonamiento lógico, simplemente porque lo que un profesor puede entregarle como material de trabajo en clase es algo tremendamente abstracto. Con el Logo el niño tiene que estructurarse mentalmente y tiene que seguir esta estructuración, de otra forma no llegará a ese fin que él pretende."



El Logo no es su única herramienta. El criticado BASIC también está presente. "Utilizamos BASIC por una razón sencilla: cuando los chicos terminen sus ocho cursos de básica y salgan a la calle, se encontrarán en cada esquina con él. Intentamos darles unas nociones muy generales, para que lleguen a poder realizar pequeños programas y prácticas dentro del contexto de las materias de ciencias que actualmente estudian." Y no sólo lenguajes, la elección de programas educativos no es menos importante: "Con los pequeños utilizamos programas pensados para que el chico se vaya familiarizando con el ordenador.

Indirectamente esto les supone toda una serie de refuerzos con el



lidad. Lo que criticamos es que el uso del aula informática quede reducido a un determinado seminario en el instituto o al profesor que conoce el tema en la escuela.

Si bien es importante que la experiencia descanse en determinados reponsables, el aula informática ofrece un instrumento técnico de gran valos para ser aprovechado por el conjunto de los profesores, sean de ciencias o de letras, sepan o no lenguajes de programación.

El proyecto pedagógico se convierte, por tanto, en la elección apropiada de un lenguaje de programación con el que trabajar, así como en la selección de los programas educativos que se utilizarán.

Para terminar, desearíamos que nos hicierais llegas vuestras experiencias sobre este tema y vuestras consultas o críticas si las hubiera.

Deseamos que esta sección contenga un correo de lectores que vinculados a la educación deseen recibir información y asistencia sobre los problemas que se puedan encontrar en la escuela trabajando con microodenadores.

Miguel Figini
Miembro del equipo de asesores de Ioshua

amigos

abecedario, mayúsculas y minúsculas, etc. Con los mayores este tipo de juegos permite desarrollar la fantasía y toda una serie de aspectos que en clase difícilmente tienen lugar."

Muchas son las experiencias que se están llevando a cabo en este terreno, y los resultados no siempre son satisfactorios. José Macías comenzó esta labor hace ya tres años, aunque no todo saldría bien al principio: "Comenzamos con cursos fuera de las horas de clase y la experiencia resultó bastante desgraciada. Es un grave error que no recomendaría absolutamente a nadie. Este no es camino para incorporar la informática a la escuela, ya que se convierte en un simple añadido.

Otro aspecto igualmente importante es la respuesta de profesores y padres. "Si no hubiera habido un mínimo de personas interesadas en el tema, difícilmente se hubiera puesto en marcha esta experiencia. El resto del profesorado está en una actitud de espera, a ver lo que sucede. Saben perfectamente que estamos en una etapa de búsqueda, de experimentación. Aunque no hayan participado en la idea de entrada, saben que puede haber una aplicación en sus diferentes campos. En cuanto a los padres, la reacción ha sido positiva al cien por cien, y lo apoyan económicamente." Una buena participación o al menos necesaria, aunque la utilización de ordenadores de bajo precio hace que no se "disparen" los costes, como puntualiza José Macías. "Es menos costoso de lo que realmente parece. Estamos intentando dar informática hasta octavo de básica y de alguna forma mentalizarlos de que el material que necesitan no es excesivamente sofisticado. Para cumplir esta misión el Plus es suficiente."

"No se puede hacer de la informática un añadido más"

"La informática tiene una importancia muy grande en el futuro de nuestra profesión y de nuestra dedicación. Yo recomiendo que lo tomen en serio y no intenten jamás hacer de ella un añadido más, como tantas actividades que se realizan fuera de las horas de clases", concluye José Macías, quien se muestra particularmente orgulloso de los logros conseguidos.



COMERCIAL

```
OFERTA-FACTURA-ALBARAN-PEDIDO

-MENU-

1- CREACION
2- MODIFICAR DATOS ARCHIVOS
3- LISTADO DATOS ARCHIVOS
4- INTRODUCIR DATOS ARCHIVOS
5- IMPRESION
6- BORRAR DATOS ARCHIVOS
7- CLASIFICACION DATOS
8- CAMBIO FORMATO DE IMPRESION
9- CAMBIO DE FICHEROS
0- DEJAR PROGRAMA

OPCION ?
```

Son cada vez más los pequeños empresarios o comerciantes que han acudido al Spectrum como ordenador con el que dar los primeros pasos en el mundo de la informática. Las razones son variadas: quizá lo compró "para que jueguen los niños", quizá porque la inversión es pequeña y "no se arriesga mucho". Para servir a este tipo de usuarios han ido apareciendo programas profesionales: contabilidades, control de stocks, etcétera.

Recientemente **ALSI** ha puesto a la venta su programa de facturación. Se trata de un programa que integra varias funciones, siendo su propósito facilitar la gestión de la pequeña empresa o comercio. Con él se puede facturar, hacer pedidos, controlar el stock, elaborar listas de precios y de clientes. Para ello el programa maneja hasta veinte ficheros diferentes, diez de clientes, proveedores o distribuidores y otros diez de artículos.

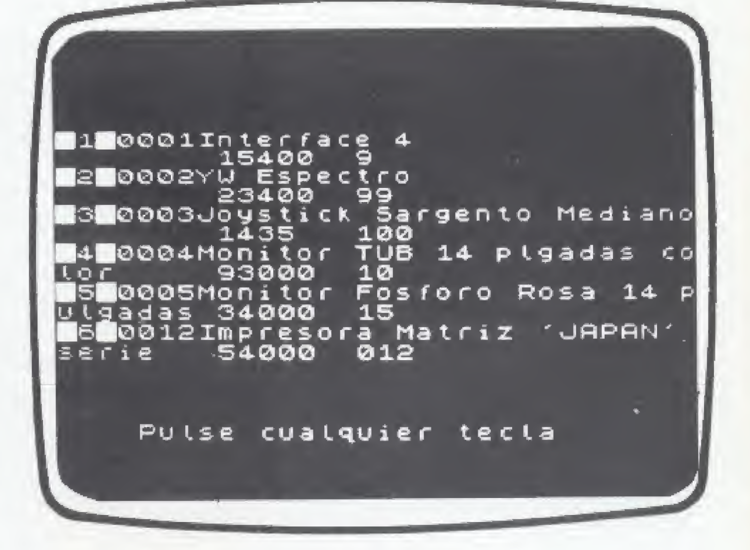
```
EL FORMATO ACTUAL ES:

=> LA "CANTIDAD" SE IMPRIME EN
   LA COLUMNA 45
=> EL "PRECIO", EN LA 53
=> EL "TOTAL", EN LA 66
=> LOS "TOTALES", EN LA LINEA 56

QUE QUIERE CAMBIAR ?

1- COL. DE IMP. "CANTIDAD"
2- COL. DE IMP. "PRECIO"
3- COL. DE IMP. "TOTAL"
4- LIN. DE IMP. "TOTALES"

5- MENU
```



Cada fichero de clientes puede manejar hasta 30, y 100 cada fichero de artículos. Siendo una aplicación donde es esencial el almacenamiento de datos, el programa trabaja en microdrive. Para imprimir facturas será necesario utilizar impresoras de 80 columnas cuyos códigos para caracteres compromidos y doble ancho coincidan con los de las impresoras Epson, pudiendo utilizar la salida RS232 del Interface 1 o un interface Centronics Kempston o Indescomp.

Tras la carga del programa por

4 ANALISIS DE SOFTWARE

primera vez, será necesario abrir, como mínimo, un fichero de artículos y otro de clientes. Además, el programa nos preguntará el nombre, dirección, teléfono, número de identificación fiscal y margen de beneficio. Estos datos serán utilizados automáticamente por el programa como cabecera de todas nuestras facturas. Después se nos pregunta el tipo de impresora que tenemos (Serie o Paralelo) y si es RS232, la velocidad de transmisión a que trabaja (en baudios).

Llegamos así al Menú Principal, que tiene diez opciones: Creación, Modificación, Listados, Introducción, Impresión, Borrado, Clasificación, Cambio de formato de impresión, Cambio de fichero y dejar el programa.

Desde este momento, podremos empezar a introducir artículos en el fichero. Por cada uno se debe introducir el código de identificación, el nombre, el número de unidades en stock y el precio de venta al público. El precio de coste será calculado automáticamente por el programa a partir del margen de beneficio.

La introducción de clientes se realiza de una manera parecida: se nos pregunta nombre, código, dirección y teléfono. Las opciones de modificación de artículos y clientes son parecidas, aunque en este caso el programa nos pregunta primero por la ficha que queremos cambiar.

Se pueden utilizar las opciones de listado de clientes y artículos para obtener copias impresas (o en pantalla) de la lista de artículos o del fichero de direcciones. La opción de clasificación permite ordenar el fichero de artículos, bien por orden numérico (de códigos) o de nombres. El fichero de clientes sólo se puede clasificar por orden alfabético.

La opción de creación nos da paso al meollo del programa: cuatro opciones que nos permiten elaborar facturas, ofertas, pedidos o albaranes. En los cuatro casos, el ordenador empieza por preguntarnos el nombre del cliente o proveedor. Si el nombre está en el fichero utiliza los datos de éste; en caso contrario nos pregunta el resto de los datos y pasa a incluirlo para referencias posteriores. Tras el nombre del cliente se nos pregunta la fecha, después se pueden incluir los códigos de referencia de hasta 10 artículos.

Si el código no está presente en el fichero de artículos, el programa nos preguntará automáticamente el resto de los datos y lo incluirá en el stock, "poniendo" un ejemplar en el almacén. Esto último no resulta deseable en muchas ocasiones. Por ejemplo, ofertamos un artículo que no tenemos en almacén. El programa nos pregunta los datos del producto. Si después hacemos un listado de existencias nos encontramos con que aparece un ejemplar.

Al acabar la lista de artículos, el programa nos pregunta la forma de pago. Esta se puede elegir entre cinco predefinidas o teclear la fórmula elegida. También se nos pide la entrega inicial (línea de texto

ALSI COMERCIAL S.A.
Antonio Lopez, 117 2-D
28026 MADRID   Tel: 4754339

OFERTA

fecha 21.11.84   num. 00006

COMPUTERLAND
CASTELLO, 89
MADRID

forma de pago: 30 % AL PEDIDO. RESTO 30, 60 Y 90 DIAS F/F.

cod.	concepto	cant.	precio	importe
360001	Articulo primero	2	11.-	22.-
370002	Articulo segundo	2	22.-	44.-
380003	Articulo tercero	2	33.-	66.-
390004	Articulo cuarto	2	44.-	88.-
400005	Articulo quinto	2	55.-	110.-

Linea de texto

Otra linea de texto

tot. bruto	%	descuento	%	i.g.t.e.	TOTAL
330.-			0.7	2.-	332.-

ANALISIS DE SOFTWARE

que podemos especificar a voluntad) y otra línea de importes y vencimientos. El ordenador rellena los totales, preguntándonos ITE y descuento. Presentamos algún ejemplo del formato de impresión del programa, con facturas, albaranes y ofertas ficticias, así como un listado de artículos y clientes de nuestra hipotética empresa.

Después de imprimir una serie de facturas, el programa nos pregunta si deseamos que permanezcan en memoria. Cuando las borremos, el programa actualizará el almacén con las salidas (facturas) o entradas (pedidos) realizadas. No existe más posibilidad, si nos hemos equivocado al realizar una factura, que comenzar de nuevo con el programa. Si lo hacemos sacando el cartucho, los cambios realizados en la sesión no quedarán reflejados en el programa.

El principal defecto del programa es la obligación de trabajar con un margen de beneficio igual para todos los artículos. Otro problema que plantea es la imposibilidad de modificar o borrar una factura incorrecta. Por lo demás, no resulta lento en su ejecución y la capacidad es acorde con los objetivos y

SUMINISTROS MARTINEZ, S.L.
Arsenal,34
Madrid   Tel: 7654321
28032 MADRID   012

OFERTA

TUMICRO
Gran Via, 51
Burgos
BURGOS

fecha   num.
12.04.85   00002

forma de pago: A CONVENIR

cod.	concepto	cant.	precio	importe
0001	Interface 4	1	15400.-	15400.-
0002	YW Espectro	1	23400.-	23400.-
0012	Impresora Matriz 'JAPAN' serie	1	54000.-	54000.-
0010	Cable Sablazo Centronics	1	5300.-	5300.-



Eugenio Ubeda es un "enamorado" de los ordenadores. Ahora sustituido por el personal de IBM, el Spectrum ha llevado sobre "sus espaldas" el control de stocks de la pequeña tienda de recambios y compra-venta de coches. El se hizo sus propios programas y podría haberse ahorrado mucho tiempo —"he invertido más de 500 horas en los programas"— si hubiese contado con un buen programa de facturación.

"El problema es que hay programas para médicos, abogados..., pero no para los automovilistas. Por eso me metí a hacer mis propios programas. Con ellos realizo la facturación de alquileres de coches, compra-venta, mailing y, sobre todo, el control de stocks. El Spectrum ha funcionado siempre bien y sin ninguna avería, pero se me había quedado pequeño, a pesar de sus 47 conceptos por artículo. El almacenamiento en cinta es uno de sus mayores probemas.

```
RELACION DE ARTICULOS AL 21.04.85    SUMINISTROS MARTINEZ, S.L.

COD.              NOMBRE                    PRECIO   CANT.    TOTAL      COSTO
0001  Interface 4                           15400.-    8    123200.-    10010.-
0002  YW Espectro                           23400.-   96   2246400.-    15210.-
0003  Joystick Sargento Mediano              1435.-  100    143500.-      933.-
0004  Monitor TUB 14 plgadas color          93000.-   10    930000.-    60450.-
0005  Monitor Fosforo Rosa 14 pulgadas      34000.-   15    510000.-    22100.-
0006  Impresora Termica BURNS               23000.-    1     23000.-    14950.-
0012  Impresora Matriz 'JAPAN' serie        54000.-   11    594000.-    35100.-
0010  Cable Sablazo Centronics               5300.-    1      5300.-     3445.-
                                                         ------------
                                                          4575400.-
```

```
003
INTERSONICA
Pedro Lopez, 25
Madrid
28003 MADRID

024
MICROTIENDA
San Fernando, 15
Alicante
03003 ALICANTE

002
MICRO-TUYO
San Pedro, 124
Toledo
TOLEDO

004
SOFT-VARE
Corts Catalanes, 84
Barcelona
08012 BARCELONA

002
TIRSA
Pi, 24
Hospitalet de Llobregat
08124 BARCELONA
```

posibilidades del Spectrum. La empresa que lo comercializa nos comentó que estaban trabajando en una versión para QL del mismo programa. Esperamos que esta versión saque partido de la superior memoria y características gráficas del QL, resultando una aplicación más acabada que la que comentamos.

Resumiendo sus principales características, un programa que nos ayuda en la gestión de almacén y facturación de pequeñas empresas y comercios. Util, aunque tenga limitaciones que ya hemos comentado. La presentación gráfica no está excesivamente cuidada, aunque donde el programa "da la talla" es en la impresora.

Pense en el QL como sustituto, pero tuve una buena oferta del PC..."

Miembro de la Asociación Nacional de Vendedores de Vehículos a Motor, se comunica habitualmente por teléfono para recabar informes sobre la legalidad o no de determinadas operaciones: "Ahora es muy lento. Tendremos que ir en el futuro a estar conectados a una base de datos..."

Después de trabajar incansablemente con el Spectrum, no duda en calificarle de una máquina extraordinaria, aunque siempre trabajaba en casa: "Lo tuve que quitar de la tienda porque a la gente no le parecía serio. Entraban y se creían que estábamos jugando a los marcianitos." Lamentablemente existe esta creencia, y aunque sea el ordenador con mayor número de juegos del mercado, también son muchos sus programas de aplicaciones.

[illegible]

[illegible]	DENOMINACION	STOCK	REFERENCIA	[illegible]	[illegible]
F01	PORTAFUSIBLE	[illegible]	--	70	[illegible]
F02	BOLSAS 2 FUSIBLES	10	464	1.90	1
F03	FUSIBLES NORMALES	274	290	10	
F04	FUSIBLES CRISTAL 10 AMP.	40	CL-10	20	14
F05	FUSIBLES CRISTAL 16 AMP.	10	CL-16	20	14
F06	LAMPARAS FARO TUNGSTENO	5	EU-21112	220	154
F07	LAMPARAS H4 HALOGENAS	2	H-4	900	630
F08	LAMPARAS 1 FILAMENTO	5	T-1036	72	50.4
F09	LAMPARAS 2 FILAMENTOS	[illegible]	T-2036	86	60.2
F10	ESTUCHE DE LAMPARAS	1	E-1003	740	518
F11	ESTUCHE DE LAMPARAS	1	E-1005	1600	1120
F12	ESTUCHE DE LAMPARAS	1	E-1011	730	511
F13	ESTUCHE DE LAMPARAS	1	E-1013	1500	1050
F14	ESTUCHE DE LAMPARAS	1	E-1014	740	518
F15	ESTUCHE DE LAMPARAS	1	E-1016	730	511
F16	ESCOBILLAS LIMPIAPARABRISAS	1	GF-3	1022	715.4
F17	ESCOBILLAS LIMPIAPARABRISAS	1	GF-4	1056	739.2
F18	ESCOBILLAS LIMPIAPARABRISAS	1	GF-5	1114	779.8
F19	ESCOBILLAS LIMPIAPARABRISAS	1	GF-6	1126	788.2
F20	ESCOBILLAS LIMPIAPARABRISAS	1	GF-7	1466	1026.2
F21	PORTATIL FIJO	1	MINI	490	343
F22	PORTATIL GIRATORIO	[illegible]	GIRATORIO	750	525
F23	TOPES DE PUERTA	4	--	310	217
F24	TAPONES DE ANTENAS	12	--	130	91
F25	COMPAPOLVOS VILEDA	27	--	100	70
F26	3 EN UNO AFLOJATODO	12	--	500	350
F27	TAPA-FUGAS RADIADORES	2	TG-250	342	239.4
F28	AUTO ELECTRIC	2	DYNAMIC	422	295.4
F29	DESENGRASANTE PARA MOTORES	1	DYNAMIC	642	449.4
F30	MULTIUSO	4	DYNAMIC	422	295.4
F31	LIMPIA VINILO	4	DYNAMIC	644	450.8
F32	LIMPIA RADIADORES	2	DYNAMIC	396	277.2
F33	BRILLO RENOVADOR	2	KRAFFT	846	592.2
F34	1 L. ANTICONGELANTE-12 C	1	CC.20%	176	123.2
F35	5 L. ANTICONGELANTE-20 C	[illegible]	CC.30%	696	487.2
F36	1 L. AGUA DESTILADA	8	--	60	42
F37	BARRA REMOLQUE	1	--	3800	2660
F38	LIQUIDO DE FRENOS	1	DYNAMIC	416	291.2
F39	LIQUIDO DE FRENOS	1	DYNAMIC STD	678	474.6
F40	LIQUIDO DE FRENOS	1	BANDERAS	656	459.2
F41	TORNILLO ANTIRROBO LLAN.	1	001	2160	1512
F42	TORNILLO ANTIRROBO LLAN.	1	002	2160	1512
F43	TORNILLO ANTIRROBO LLAN.	1	003	2160	1512
F44	ANTIRROBO RUEDA P-205	7	--	[illegible]	[illegible]
F45	TAPACUBOS LLANTA SEAT	1	--	400	280
F46	TAPACUBOS LLANTA SIMCA	1	--	450	15

[illegible]

⑤

Descubrimiento de un nuevo lenguaje: PASCAL

Una vez vistas las variables de tipo estándar dentro del PASCAL, vamos a hablar en esta ocasión de otro tipo de datos y variables, las TYPE, tanto simples como compuestas.

Dentro de las simples tenemos los tipos enumerados o escalares, los conjuntos y el tipo subrango o intervalo. Al hablar de las compuestas nos estamos refiriendo a las matrices, llamadas en PASCAL *arrays*.

Variables TYPE simples

Al igual que los demás tipos ya vistos, es necesario declararlas en su lugar correspondiente. Si nos fijamos en el diagrama general de un programa Pascal, visto en capítulos anteriores, observamos que este lugar está entre la declaración de constantes y variables y que deben ir precedidas por la palabra reservada TYPE.

Tipos enumerados o escalares

Este lenguaje, aparte de proporcionar los tipos de variable estándar, permite definir nuestros propios tipos de datos, conseguimos esto mediante el siguiente tipo de declaración:

```
TYPE
   T = (A1, A2, ..., An);
```

En los siguientes ejemplos indicamos que la variable juegos puede tomar cualquiera de los cuatro valores que se indican, pero nunca otro de los no especificados. Lo mismo ocurre con la variable marcas.

```
TYPE
   juegos = (tute, mus, canasta,
   poker);
   Marcas = (renault, seat, ci-
   troën, volvo);
```

Si queremos utilizar estos tipos dentro de un programa, tendremos que declarar variables del tipo previamente definido. Así por ejemplo:

```
VAR
   mi juego: juegos;
```

Dada esta declaración las asignaciones:

```
mi juego: = mus; o mi juego: =
tute;
```

serían perfectamente válidas. (Ver diagrama sintáctico en figura 1.)

Otro aspecto importante a resaltar son los operadores que podemos utilizar con las variables de tipo escalar. Solamente está permitido utilizar los operadores de relación, obteniendo como resultado de las expresiones un valor booleano que como sabemos toma los valores TRUE o FALSE.

El orden de los elementos dentro de una lista de tipos enumerados está definido por la posición en que éstos se colocan en la declaración. Así, para nuestro ejemplo, el primer elemento será tute, el segundo mus...

Expresiones perfectamente válidas dentro de un programa podrían ser:

```
tute < mus, citroen > volvo,
```

dando como resultado de la evaluación TRUE y FALSE, respectivamente.

Como ya vimos al estudiar los datos de tipo carácter, existen tres funciones incorporadas del Pascal que nos pueden ser muy útiles para trabajar con este tipo de datos. Estas funciones son PRED, SUCC y ORD.

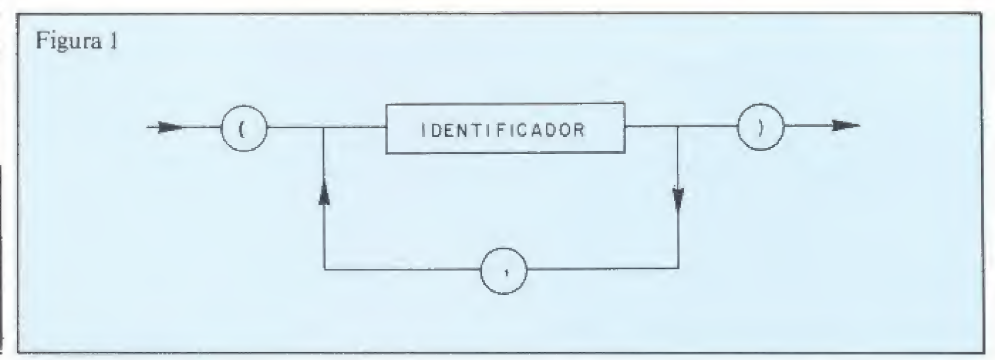
Recordemos que las funciones PRED y SUCC operan con tipos escalares y devuelven el elemento anterior y posterior, respectivamente, dentro de la lista de elementos de la declaración, mientras que la función ORD nos devuelve un valor entero, que indica la posición del elemento en cuestión dentro de dicha lista. Hay que tener en cuenta el primer elemento de la lista que tiene el valor ordinal 0, y no 1, como cabría suponer.

En nuestros ejemplos:

```
ORD (mus) = 1
PRED (poker) = canasta
SUCC (seat) = citroen
ORD (citroën) = 2
PRED (seat) = renault
etc.
```

Tipo subrango

También llamado intervalo. Mediante este tipo podemos declarar variables que están siempre dentro de un margen, dando el límite superior e inferior y definiendo así un intervalo.



La forma de definición es:

```
TYPE
    T = liminf..limsup;
```

siendo *liminf* y *limsup* constantes.

Estas constantes que delimitan el intervalo pueden ser de tipo *integer, char* o del tipo escalar visto anteriormente, pero nunca de tipo real. (Ver diagrama sintáctico en figura 2.)

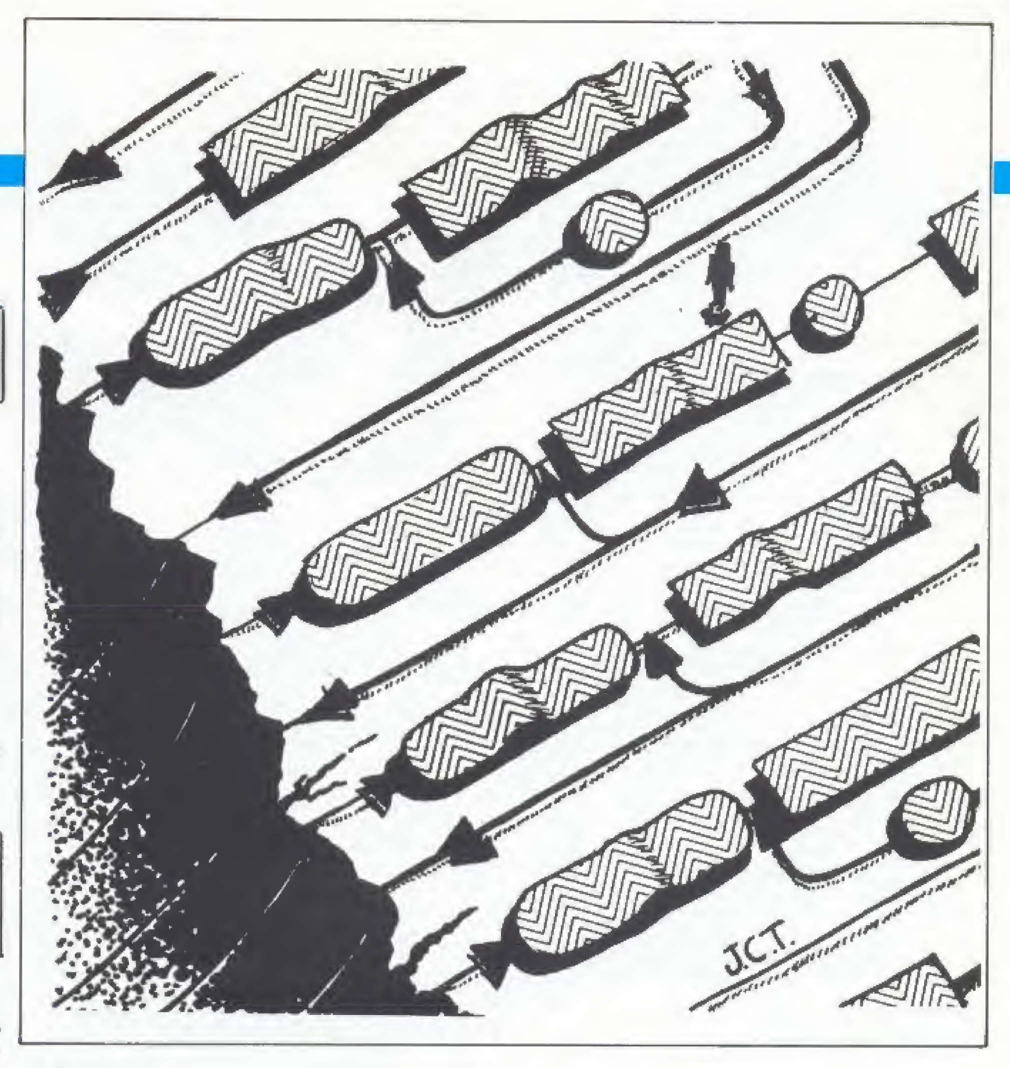
Ejemplos de declaraciones de datos de tipo subrango son:

```
VAR
    dígito: '0'..'9';
    laborables: lunes.. viernes;
```

Para esta última declaración, previamente deberíamos haber definido el tipo de datos día-semana de la siguiente forma:

```
TYPE
    díasemana = (lunes, martes,
    miércoles, jueves, viernes,
    sábado, domingo);
```

Existe un pequeño problema con el manejo del tipo subrango: salirse de los límites impuestos. Aunque en principio no sea un error sintáctico el error aparecerá en tiempo de ejecución. Veámoslo con un ejemplo:




```
TYPE
    intervalo = 1.. 10;
VAR
    número: intervalo;
```

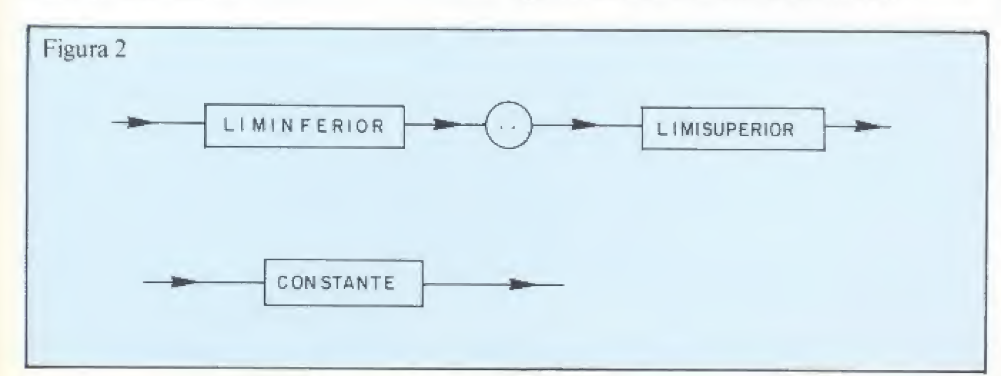
La asignación número: = 11 es perfectamente válida sintácticamente, y sin embargo al ejecutar el programa aparecerá un error al salirnos del rango impuesto para la variable número.

Para aclarar lo expuesto el programa 1 imprime todos los números de tres cifras de forma que la suma de los cubos de las cifras son iguales al número en cuestión.

Programa 1

```
PROGRAM cubo;
TYPE
    dígito = 0.. 9;
VAR
    i, j, k: dígito;
BEGIN
  FOR i: =O TO 9 DO
  FOR j: = O TO 9 OO
  FOR k: = O TO 9 DO
      IF i * i * i + j * j * j + k * k *
      k = 100 * i + 10 * j + k
      THEN writeln (i : l, j : l, k
      : l)
END.
```

Figura 2



Tipo SET o conjunto

Todos sabemos lo que es un conjunto en matemáticas. Lo podemos definir como una serie de objetos del mismo tipo, pero a la vez distintos entre sí. Los elementos pertenecientes a dicho conjunto se dice que son sus miembros. (Ver diagrama sintáctico en figura

3.) El formato general del conjunto en Pascal es:

```
TYPE
    identificador = SET OF
    identificador;
```

Matemáticamente, un conjunto se indica encerrando sus elementos entre llaves. En Pascal, por estar éstas reservadas a los comentarios, indicaremos un set encerrando entre corchetes todos los elementos que lo componen.

Para aclarar las operaciones que se pueden realizar con los conjuntos haremos las siguientes declaraciones previas:

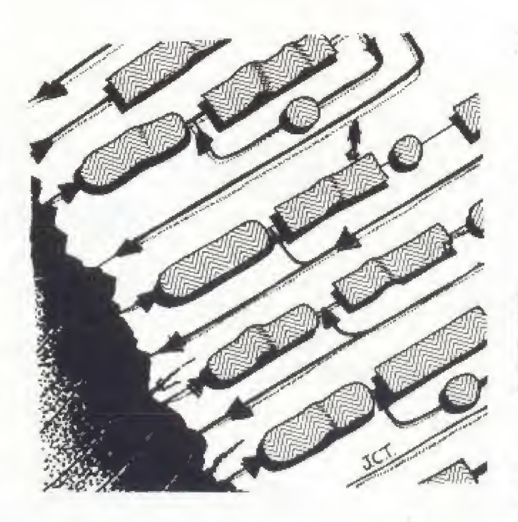
```
TYPE
    colores = (rojo, violeta, azul,
    gris, marrón);
    colorpintura = SET OF co-
    lores;
VAR
    colorcoche: colorpintura;
```

Conjuntos de nuestro ejemplo serán:

```
[azul, gris]
[rojo, marrón, azul]...
```

Si un SET no tiene miembros se dice que está vacío (SET o conjunto vacío) y su representación es: []

Terminada esta introducción a la teoría de los conjuntos, entramos a ver las distintas operaciones que podemos realizar con este tipo de datos. Las básicas son: unión, intersección, diferencia, pertenencia e inclusión. Veamos detenidamente cada una de ellas:



Unión

La unión de dos conjuntos representa otro conjunto, el cual contiene los miembros de ambos. Matemáticamente esta operación se representa como $A \cup B$. En lenguaje Pascal, utilizaremos el operador '+'.

Un ejemplo de unión podría ser:

```
[marrón] + [gris, verde] = [ma-
rrón, gris, verde]
```

Intersección

La intersección de dos conjuntos (A B) es otro conjunto que contiene los elementos que son miembros de ambos (A y B). El operador de intersección para nosotros es '*'. En los ejemplos antes definidos:

```
[marrón, gris, violeta] * [azul,
violeta, rojo] = [violeta]
```

Diferencia de conjuntos

La diferencia entre dos conjuntos A y B es otro conjunto, C, que contiene los elementos pertenecientes a A que no están en B. Por ejemplo:

```
[azul, marrón gris] - [gris, viole-
ta] = [azul, marrón]
```

Es decir, son los elementos del primer conjunto que no están en el segundo.

Pertenencia a un conjunto

Utilizaremos la palabra reservada IN para indicar la pertenencia de un elemento a un conjunto. El resultado de esta operación o evaluación será siempre TRUE o FALSE.

Por ejemplo:

```
'7' IN ['0'..'9'] al evaluarlo dará
como resultado TRUE.
```

Operadores de relación para comparar conjuntos

A = B	indica igualdad entre los conjuntos A y B.
A < >B	indica desigualdad entre ellos.
$A \geqslant B$	indica que A contiene a B.
$A \leqslant B$	indica que A está incluido en B.

El resultado de todas estas expresiones no puede ser otro que TRUE o FALSE.

Para nuestros ejemplos:

[rojo, marrón, violeta] ⩾ [violeta, rojo] es TRUE, ya que el primer conjunto contiene todos los elementos del segundo.

El programa 2 vacía un conjunto de elementos, llenando otro con todos los números primos contenidos entre dos y treinta.

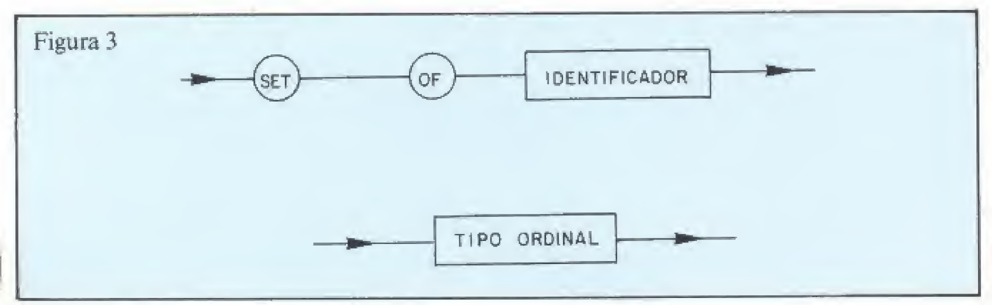


Figura 3

Exclusivo para lectores de **Todospectrum**

CLUB DE SOFTWARE

Ha seleccionado ocho excelentes programas de juego para su ordenador SPECTRUM



Precio: 1.550 ptas.

BLACK HAWK

Usted es un piloto de pruebas altamente entrenado para pilotar el avión más destructivo hasta ahora creado. Dos pantallas con casi 30 niveles de habilidad.



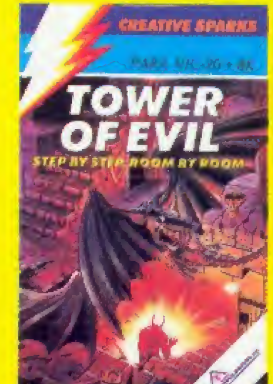
Precio: 1.550 ptas.

DANGER MOUSE

El malvado Barón Silas Greenback ha construido un androide que Danger Mouse debe destruir sin perder un momento.

Guie a Danger Mouse y a Penfold a través de la espesa jungla hasta el lugar donde se encuentra el Barón, y después deténgale antes de que sea demasiado tarde.

¿Puede usted ayudar a Danger Mouse a salvar el mundo?



Precio: 1.550 ptas.

TOWER OF EVIL

¡Arriésguese!
¿Puede usted recuperar el tesoro perdido? Una acción que discurre a través de 40 o más habitaciones le aseguran horas de juego que cautivará su atención.

- Número de jugadores uno
- Control joystick o teclado
- Dos pantallas de gráficos de acción rápida
- El VIC-20 requiere un cartucho 8K



Precio: 1.550 ptas.

DELTA WING

Delta Wing es un avanzado simulacro de combate en vuelo. Ha sido diseñado para dar la excelente "sensación" de los giros y volteretas de la alta velocidad en vuelo. Con 15 controles independientes y aviones de ataque enemigos para derribar, este programa le lleva a usted y a su Spectrum al límite.

Delta Wing también permite conectar dos ordenadores Spectrum para jugar uno contra otro mediante el interface 1.



Precio: 1.550 ptas.

SPECIAL DELIVERY

Santa Claus se encuentra metido en un terrible lío: se durmió en los laureles, y ahora no tiene suficientes regalos para visitar a todos los impacientes niños.

Ayude a Santa Claus a recoger regalos adicionales y después a descender sobre los tejados para entregar los regalos introduciéndose por las diversas chimeneas.

- Tres pantallas rodantes.
- Amplia animación gráfica y música.



Precio: 1.550 ptas.

RIVER RESCUE

Un equipo de importantes científicos han quedado atrapados en una peligrosa jungla y la única manera de sacarlos de allí es a través del río. Usted es el propietario de una potente fueraborda y su trabajo es rescatarlos.

Navegar a través de uno de los más salvajes y tortuosos ríos del mundo no resulta fácil, pero usted debe dirigir su magnífica nave corriente arriba, sorteando pequeñas islas, evitando cocodrilos y leños flotantes a su paso. Disfrute la rápida y excitante acción en este juego de alta energía. La acción en la pantalla rodante y el magnífico sonido y gráficos le aseguran largas horas de diversión.

Recorte y envíe este cupón HOY MISMO a: **INFODIS, S.A. Bravo Murillo, 377-5.º-A. 28020 MADRID**



Precio: 1.550 ptas.



Precio: 1.550 ptas.

ROAD RACER

Adelanta al primer coche y a otro más, su pulso se acelera y la tensión crece, ¿podrá tomar la delantera y ganar el trofeo?

GOLD RUSH

La emocionante aventura de la búsqueda de oro en un extraño planeta llamado Orón. ¿Qué peligros le aguardan...?

ENVIE A MI DOMICILIO

LA CASSETTE O CASSETTES RELACIONADAS A CONTINUACION EL IMPORTE LO ABONARE:

POR CHEQUE ☐ CONTRA REEMBOLSO ☐ CON TARJETA DE CREDITO ☐
American Express ☐ Visa ☐ Interbank ☐ Fecha de caducidad ____

Número de mi Tarjeta ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Nombre ____

Dirección ____

Ciudad ____ D.P. ____

Provincia ____

cantidad	producto	ptas.	total

SIN MAS GASTOS DE ENVIO

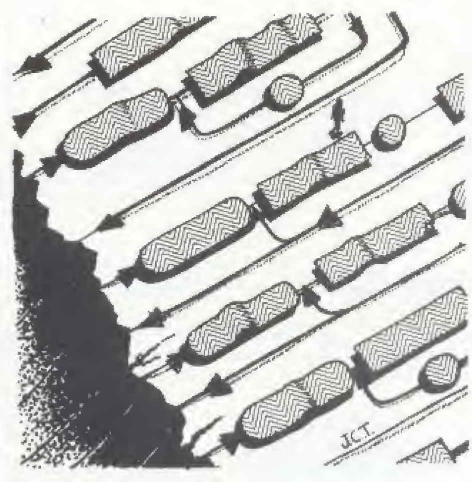
Programa 2

```
PROGRAM CONJUNTO;
CONST
   N=30;
TYPE
   ENT=1..30;
VAR
   PR, CR: SET OF ENT;
   NUM: ENT;
   I: INTEGER;
BEGIN
   PR:=[];
   CR:=[2..N];
   NUM:=2;
   REPEAT
     WHILE NOT (NUM IN
     CR) DO NUM:=NUM+
     1;
     PR:=PR+[NUM];
     WRITE(NUM);
     I:=NUM;
     WHILE I ≤ N DO BE-
     GIN
       CR:=CR-[I];
       I:=I+NUM
      END;
     UNTIL CR=[]
END.
```

Para terminar con este capítulo, veremos la última sentencia de control que nos queda, y que utiliza mucho las variables TYPE, de las que hemos venido hablando.

Sentencia CASE

Mediante la sentencia IF, teníamos la posibilidad de elegir una de



entre dos alternativas posibles. Pero puede ocurrir que en un programa deseemos elegir una alternativa entre varias (más de dos). En estos casos será necesaria la sentencia CASE. (Ver diagrama sintáctico en figura 4.)

El formato general de la sentencia es:

```
CASE expresión OF
e1: tratamiento 1;
e2: tratamiento 2;
. . . . . . . . . . . . . . . . . . . . . . . . . .
en: tratamiento n;
END.
```

Figura 4

CASE → EXPRESION → OF → CONSTANTE → : → SENTENCIA → END (,) (;)

De esta forma, CASE selecciona aquella sentencia cuya constante sea igual al valor actual de la expresión. Si no existiera la etiqueta correspondiente, el efecto es indefinido. Algunos compiladores tienen la opción OTHERS para el caso en que la etiqueta dada no exista. Con un ejemplo entenderemos mejor esta sentencia: El programa 3 efectúa una de las cuatro operaciones básicas, dependiendo del valor de una variable.

Programa 3

```
PROGRAM OPERACIONES;
TYPE
   LIMITE=Ø..4;
VAR
   OP: LIMITE;
   A, B, R : REAL;
BEGIN
   READ (A, B, OP);
   CASE OP OF
     0: R:=A B;
     1: R:=A-B;
     2: R:=A B;
     3: R:=A DIV B;
     4: R:=A/B;
     END;
     WRITE ('EL RESULTA-
     DO ES: ', R)
END.
```

En el próximo capítulo hablaremos sobre las matrices dentro del Pascal, así como de los registros.

José R. Herreros



SERVICIO DE EJEMPLARES ATRASADOS

Complete su colección de

Todospectrum

A continuación le resumimos el contenido de los ejemplares aparecidos hasta ahora.

Núm. 1 • 250 pts.
Cómo usar el microdrive/Programación Basic/Ampliación Basicare/Rutina despertador/Variables del sistema/Entrada datos mediante máscaras/Protección del software/Sintonice su Spectrum/Programas.

Núm. 2 • 250 pts.
Gráficos profesionales/Desplazamiento pixel a pixel/Utilización de rutinas/Construcción del interface centronics/Programas de utilidad para microdrive/Rutina reset en código máquina/Análisis del editor de textos Tasword/Interfaces para impresoras/Programas.

Núm. 3 • 250 pts.
Novedades sonimag '84/Ampliando el Basic/Programas para ordenar programas/Gráficos con el VU-3D/Lenguaje Forth/Archivos en microdrive/Programación de un interface de impresora/Programas.

Núm. 4 • 250 pts.
De profesión: programador/Consola para el Spectrum/Comparación código máquina-Basic/Análisis programa contabilidad /Calendario/Pascal/Programas.

Núm. 5 • 250 pts.
Floppys para Spectrum/Diseño asistido por ordenador/64 Caracteres por línea/Juego de la vida/Pascal/Asi hacemos las portadas/Control de evaluaciones/Programas.

Núm. 6 • 250 pts.
Representación de funciones/Todos los caminos conducen a la ROM/Juegos/Pascal/Construcción de un lápiz óptico/Programas de gestión. El SITI/Logo: tortugas para todos/Interrupciones del Z-80/Programas.



Todospectrum
Todospectrum
Todospectrum
GHOSTBUSTERS
Lo mejor en representación de funciones
LOGO: Tortugas para todos
Interrupciones del Z80

DISPONEMOS DE TAPAS ESPECIALES PARA SUS EJEMPLARES DE

Todospectrum

SIN NECESIDAD DE ENCUADERNACION





(cada tapa es para 6 ejemplares)

Para hacer su pedido, rellene este cupón HOY MISMO y envíelo a:

Todospectrum
Bravo Murillo, 377
Tel. 733 96 62 - 28020 MADRID

Ruego me envíen los siguientes ejemplares atrasados de TODOSPECTRUM .... al precio de 250 pts.

Por favor envíenme .... tapas para la encuadernación de mis ejemplares de TODOSPECTRUM, al precio de 600 pts. más gastos de envío.

El importe lo abonaré
☐ POR CHEQUE ☐ CONTRA REEMBOLSO ☐ CON MI TARJETA DE CREDITO ☐ AMERICAN EXPRESS ☐ VISA ☐ INTERBANK

Número de mi tarjeta: ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Fecha de caducidad .... Firma

NOMBRE ....

DIRECCION ....

CIUDAD .... C. P. ....

PROVINCIA ....

WAFADRIVE

¿Alternativa al interface 1?

Entre la lentitud del *cassette* y los precios exorbitantes de las unidades de *diskettes,* algunas casas de *hardware* han trabajado en una alternativa de compromiso.

El *Wafadrive* es el resultado de los esfuerzos de **Rotronics** para conseguir un medio de almacenamiento con mejores características que el *cassette.* Una caja de color negro encierra dos unidades de almacenamiento, una salida Centronics y un *interface RS232. Los drives* utilizan un sistema de cinta de video en bucle sin fin, con una cabeza de *cassette* (monaural) y capacidades de 16, 64 y 128 K después de formatear.

A los lectores que conozcan el *microdrive* les resultará familiar la descripción del *Wafadrive:* existe



un gran parecido entre los dos sistema de almacenamiento. El procedimiento de ampliar el BASIC es el mismo en ambos, los nuevos comandos son muy parecidos y el sistema de almacenamiento es el mismo. Los dos periféricos son incompatibles, ya que utilizan el mismo sistema de paginación.

Si bien la velocidad de acceso no es muy grande, el sistema tiene grandes ventajas sobre el *cassette:* el ordenador realiza por nosotros la operación de localizar el principio del fichero, cargarlo o grabarlo y verificarlo. Veamos los principios generales de funcionamiento de este periférico.

El primer lugar, la cinta debe ser «formateada». El ordenador la divide en pequeños bloques, en cada

Y de regalo...
un procesador de textos

Siguiendo la tendencia de regalar programas a los compradores de ordenadores y periféricos, Rotronics adjunta al *Wafadrive* un cartucho de 16 K que contiene un procesador de texto *(Spectral Writer).* Un programa de utilidad que nos ayuda a redactar trabajos, cartas o documentos. El ancho de los documentos es de 64 caracteres, que permite la impresión a 80 columnas dejando un margen de 8 caracteres a ambos lados del texto; tiene, en cambio, el inconveniente de que los caracteres son muy poco legibles en pantalla, al tener sólo 4 *pixels* de ancho.

La principal ventaja del *Spectral Writer* sobre otros procesadores de texto para Spectrum es su diseño a la medida: al haber sido construido pensando específicamente en los *interfaces* del *Wafadrive,* el usuario no tiene necesidad de "parchear" el programa para que funcione con su impresora. Por esta misma razón, el código es muy compacto y permite documentos más largos (352 líneas de 64 caracteres) que otros programas similares.

Para introducir texto basta teclearlo en la posición deseada. El

Comandos Extendidos del Wafadrive

CAT * "d:"
Muestra el catálogo del cartucho d.

CAT # "d:"
Carga en memoria el directorio del cartucho d, sin sacarlo por pantalla.

CLEAR *
Cierra todos los canales abiertos.

CLOSE # flujo
Cierra el canal especificado. Si el número está entre 0 y 3, le devuelve su valor por defecto.

CLOSE # * flujo
Cierra el canal especificado. Si estaba abierto a un archivo, éste se escribe al cartucho y se actualiza el directorio.

CLS *
Borra la pantalla y pone tinta negra, papel blanco y borde blanco.

ERASE * "d: nombre"
Borra un archivo del cartucho; se puede especificar el nombre acabado en * para borrar los archivos cuyo nombre comienza por las letras indicadas.

FORMAT * "R"; velocidad



uno de los cuales guarda 1 K de información. Una vez realizada la división, el *Wafadrive* escribe y lee cada sector con datos aleatorios. Si la lectura es errónea, marca el sector como defectuoso y no lo utiliza más. Una vez realizada esta operación, la cinta queda en condiciones de uso. La ventaja de dividir en bloques el cartucho es que, de esa manera, es relativamente fácil localizar el principio de un sector. Además, no hay peligro de destruir información en un bloque posterior si reescribimos un sector a una velocidad ligeramente inferior. El mayor inconveniente consiste en que un archivo tiene menos de 1 K, ocupará 1 K en la cinta, ya que esta es la unidad mínima de información. En el caso del *Wafadrive*, el tiempo que tarde en localizarse un sector es, en el peor de los casos, de 6,5 segundos para cartuchos de 16 K y de 45 segundos para los de 128 K. Lógicamente, serán mucho menores en un caso promedio. Una vez localizado el

programa escribe sobre el texto presente en pantalla. Si queremos insertar en el texto tendremos que abrir un hueco en la línea deseada. Existen dos comandos para ello: uno para añadir espacios en blanco en una línea y otro para separarla en dos.

Otros comandos nos permiten avanzar a lo largo del documento, posicionarnos a final de párrafo, justificar una línea o un párrafo entero. El programa se encarga de que las palabras nunca queden a caballo entre dos líneas, y realiza, por defecto, la justificación del texto insertando espacios que ajusten el margen derecho.

Junto a los comandos de edición existe la posibilidad de salvar, cargar o mezclar textos a *Wafadrive* o *cassette*, además de un comando de

Fija la velocidad del RS232.

FORMAT * "d: nombre"
Formatea un cartucho con el nombre indicado. Borra todos los ficheros.

INKEY $ # flujo
Lee un carácter del canal y devuelve su valor.

INPUT # flujo; variables
Lee variables del canal.

LIST # flujo, línea
Lista el programa, empezando en la línea indicada, al canal indicado.

LOAD * "d: nombre"
Carga el programa especificado; puede ser BASIC o código máquina. LOAD * carga el primer programa de la unidad por defecto.

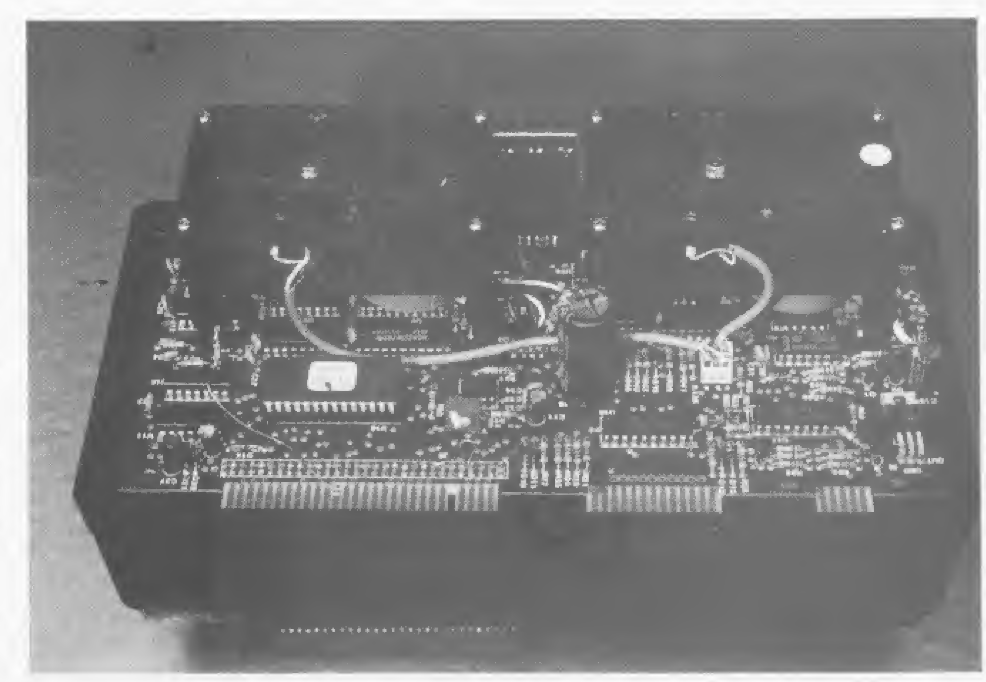
MERGE * "d: nombre"
"Mezcla" el programa BASIC indicado con el que se encuentre en memoria.

MOVE * "d: nombre 1" TO "d: nombre 2"
Copia el fichero 1 en el 2; el uso de * como nombre permite la copia del cartucho entero.

NEW
Borra el prgrama BASIC y desconecta el

sector, la transferencia de datos se hace a 2 K caracteres por segundo, aproximadamente.

Una vez formateado, cada cartucho tiene un archivo especial, llamado directorio. En él se almacenan los nombres, tamaños y números de sector ocupados por los archivos existentes en el *Wafadrive*. Este admite hasta 82 ficheros distintos (16 en cartuchos de 16 K). El directorio es leído de la cinta al empezar a trabajar, y se actualiza cada vez que se va a escribir un sector.



Los comandos de manejo de archivos se basan en los del *cassette*, seguidos de un asterisco para diferenciarlos de las órdenes referidas a éste. El nombre del fichero es de la forma «A: nombre», o «B: nombre», donde A o B se refieren a las dos unidades. Si no se indica la unidad el archivo será buscado en la unidad por defecto (normalmente A). Dos comandos extras son ERASE Y MOVE; el primero borra un fichero y el segundo copia un archivo en otro. El comando ERASE *«A:ABC*» borra todos los archivos de la unidad A cuyo nombre comience por ABC. Los comandos citados se encargan de la grabación de programas y CODE. La opción de salvar variables (existente en el *cassette)* no ha sido implementada en el *Wafadrive*, aunque puede simularse definiendo esa variable en ausencia de programa y salvándola como programa. La llamada mediante MERGE la cargará de nuevo a memoria.

Para el manejo de ficheros incluye los comandos OPEN # y CLOSE#, que se utilizan en conjunción con PRINT#, INPUT#e INKEY$#. El comando OPEN#abre

búsqueda de palabras. Sólo echamos en falta una opción de búsqueda y sustitución de palabras.

Se pueden obtener copias con la ZX Printer o compatibles, así como con cualquier impresora Centronics o RS232 (quizá con algún problema de cableado, ya que los conectores del *Wafadrive* no son estándar); los caracteres gráficos de bloques del Spectrum pueden ser utilizados para el envío de códigos de control a la impresora.

Sin llegar a la calidad de equipos profesionales, fuera de las pretensiones de un ordenador como el Spectrum, este procesador de textos nos ofrece la posibilidad de mejorar la presentación de nuestras cartas y documentos, siempre, claro está, que dispongamos de una impresora. Una muestra de que los usuarios, además de juegos, quieren que su ordenador sea una herramienta de trabajo.

Wafradrive, liberando la memoria reservada.

NEW *
Inicializa el *Wafadrive*, reservando alrededor de 2 K para el sistema.

NEW #
Borra el programa BASIC sin desconectar el *Wafadrive*.

OPEN # flujo, "canal"
Abre y asocia el flujo y el canal mencionados.

OPEN # * flujo, "Port"
Abre el flujo especificado y lo asocia al port mencionado.

OPEN # * flujo, "d: nombre"
Abre el flujo mencionado y lo asocia al fichero indicado.

PRINT # flujo; "cadena", datos, variables
Envía los datos indicados al flujo de salida mencionado (previamente abierto).

SAVE * "d: nombre" LINE número
Salva al fichero especificado el programa BASIC.

SAVE * "d: nombre", comienzo, longitud, dirección
Salva código máquina. Opcionalmente se puede especificar una dirección para autoejecución.



Las capacidades del *Wafadrive* no se limitan a sus cartuchos: la unidad incluye también un *interface* RS232 y otro Centronics. El primero es un *interface* serie muy utilizado para conexiones con modems e impresoras, y es bidireccional; el segundo es paralelo, sólo de salida y se utiliza para controlar impresoras. Se pueden abrir canales a los dos: OPEN # * 3, «c», y OPEN # * 3, «n», respectivamente. El RS232 requiere un FORMAT que especifique su velocidad de transmisión, ya que puede funcionar a casi todas las velocidades standard.

un canal a un fichero. Si éste ya existía, será abierto para lectura, y para escritura en caso contrario. Se puede imprimir o hacer INPUT de los archivos con sólo especificar el número de canal. Por ejemplo...

```
OPEN # * 4, «a test»
PRINT # 4; 1000
CLOSE # *4
```

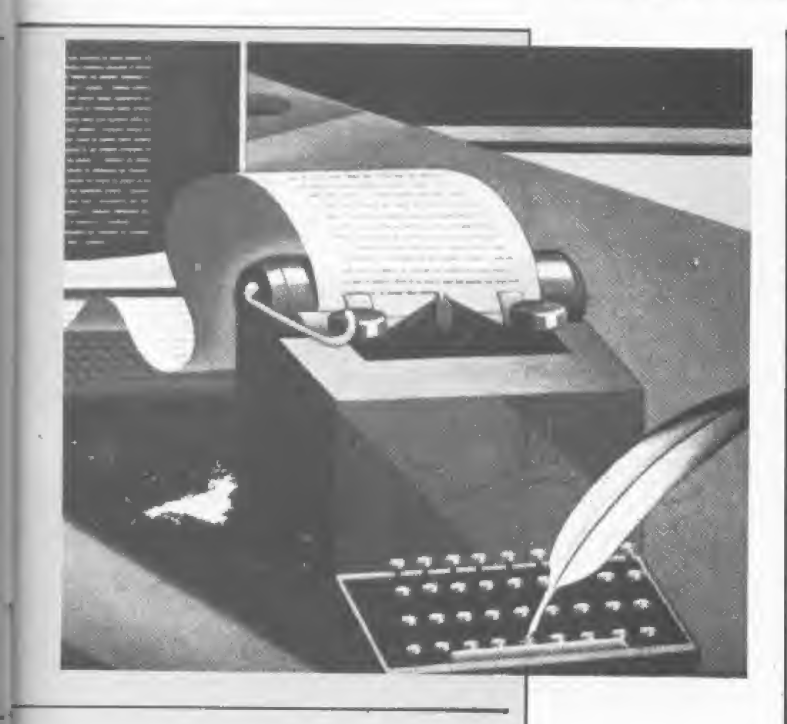
Crea un archivo y graba el número 1000.

Posteriormente:

```
OPEN # * 4, «a test»
INPUT # 4; n
CLOSE # * 4
```

Da a «n» el valor 1000.

Las comparaciones, aunque odiosas, son inevitables, y algunos lectores que conozcan el *Interface* I se preguntarán qué ventajas reales aporta el *Wafadrive* sobre el producto de Sinclair. En primer lugar, podemos citar la presencia del *Port* Centronics, que falta en el *Interface* 1, y los dos *drives*. Por otro lado, el operativo del *Wafa-*



SAVE #
Usado en lugar de SAVE *, sobreescribirá automáticamente el archivo si existía uno con el mismo nombre.

VERIFY * "d: nombre"
Verifica programas o código máquina.

drive resulta más potente que el del *Interface* 1 en el manejo de archivos.

Los puntos fuertes del *Microdrive* son la red local y la modularidad (de 1 a 8 *drives*); resulta, además, algo más rápido, ya que utiliza una cabeza estéreo, grabando dos bits cada vez. El catálogo, en cambio, es mucho más rápido en el *Wafadrive* que en el *Microdrive*, ya que este último no guarda el directorio en memoria. Por contra, la cantidad de RAM reservada por el operativo es bastante mayor en el *Wafadrive* que en el *Interface* 1.

En resumen, ninguno de los dos sistemas resulta definitivamente superior al otro y el precio tampoco marca grandes diferencias. Quizá la oferta de un cartucho con un procesador de textos por parte de **Rotronics** pudiera decantar hacia su lado a algún indeciso...

«Copys» de pantalla

Realizada la inversión de equipar tu Spectrum con un *interface* tipo paralelo y una impresora de matriz de puntos, es muy probable que seas incapaz de obtener copias de la pantalla. Es una situación bastante desagradable, porque incluso la **ZX Printer** puede hacerlo. Afortunadamente, si tu impresora es capaz de operar en modo de imagen punto por punto *(bit image mode)* todo lo que necesitas para conseguirlo es este programa en código máquina. Ha sido escrito para usarlo con el *interface* Centronics del *Wafadrive,* pero puede adaptarse a otros *interfaces.*

La rutina enviará una copia de la pantalla (incluyendo las dos últimas líneas) a una impresora de matriz de puntos ajustada para operar en modo de imagen punto a punto. En las **Epson MX70/MX80** producirá una imagen en densidad normal de 108 mm de ancho, pero usando las **Epson RX80/FX80/100** puede conseguirse además una imagen de 162 mm usando el modo *CRT Graphics.* Con las impresoras **Epson** compatibles no debería haber problemas, pero en cualquier caso quedan suficientes espacios en blanco en el listado para permitir cambios o inserciones. No obstante, si la impresora tiene menos de ocho puntos en su cabeza de impresión, habrá que transformar el programa radicalmente.

La rutina ha sido ensamblada para un Spectrum de 48 K, con la seguridad de que pocos o ninguno de los propietarios del *Wafadrive* tendrán la versión de 16 K. Su funcionamiento es bastante lento, puesto que la impresora necesita los datos en un formato totalmente diferente al usado por el archivo de imagen del Spectrum. Utiliza los 256 bytes del *buffer* de impresora del Spectrum como área de almacenamiento temporal. Las instrucciones del comienzo del listado no son demasiado concisas (muchos CALL PRINT), pero al menos son claras y es más fácil ver lo que hacen por si fuera necesario adaptarlas a una impresora no Epson. De la dirección FE6Ah en adelante, el código no necesitará ser modificado, a menos que vayamos a reubicarlo.

No es necesario que el *Wafadrive* esté activado para usar la rutina de copia de pantalla. Pero supongamos que lo esté. Después de hacer CLEAR 64975, introducir el código usando un cargador hexadecimal. Grabar el programa con SAVE * "a: dump", 64976, 0343 y luego verificarlo mediante VERIFY * "a: dump" (la sintaxis del *Wafadrive* no necesita la palabra CODE).

Es interesante fijarse en la subrutina PRINT, ya que usa las extrañas instrucciones que requiere el *Wafadrive.* Como explica el manual, las instrucciones IN se utilizan incluso para enviar datos al *port* paralelo. El libro da ejemplos de cómo hacerlo en BASIC. Aquí las instrucciones están traducidas al código máquina. Así:

```
LD B, A
LD C, 14
IN A, (C)
```

```
FDD0                      ORG   0FDD0H
5B00               BUFFE  EQU   23296
FDD0 C36AFE        S      JP    SING
FDD3 CD3FFE        EIGHT  CALL  PRINT
FDD6 3E1B                 LD    A,1BH
FDD8 CD3FFE               CALL  PRINT
FDDB 3E41                 LD    A,"A"
FDDD CD3FFE               CALL  PRINT
FDE0 3E08                 LD    A,8
FDE2 CD3FFE               CALL  PRINT
FDE5 CD36FE               CALL  CR_LF
FDE8 C9                   RET
FDE9 00                   DEFB  0,0,0,0,0
     00 00 00 00
FDEE 00                   DEFB  0,0,0,0,0
     00 00 00 00
FDF3 3E1B          NORMA  LD    A,1BH
FDF5 CD3FFE               CALL  PRINT
FDF8 3E32                 LD    A,"2"
FDFA CD3FFE               CALL  PRINT
FDFD CD36FE               CALL  CR_LF
FE00 C9                   RET
FE01 00                   DEFB  0,0,0,0,0
     00 00 00 00
FE06 C5            SINGL  PUSH  BC
FE07 3E1B                 LD    A,1BH
FE09 CD3FFE               CALL  PRINT
FE0C 3E4B                 LD    A,"K"
FE0E CD3FFE               CALL  PRINT
FE11 00                   DEFB  0,0
     00
FE13 00                   DEFB  0,0,0
     00 00
FE16 3E00                 LD    A,00H
FE18 CD3FFE               CALL  PRINT
FE1B 3E01                 LD    A,01H
FE1D CD3FFE               CALL  PRINT
FE20 21005B               LD    HL,BUFFE
FE23 0600                 LD    B,0
FE25 7E            PLP    LD    A,(HL)
FE26 CD3FFE               CALL  PRINT
FE29 23                   INC   HL
FE2A 10F9                 DJNZ  PLP
FE2C CD36FE               CALL  CR_LF
```

pasa a la pág. 39



QL
MAGAZINE
Suplemento especial Mayo 1985
EASEL
LO MEJOR DE PSION
Listado
MUSICA CON EL QL
NOVEDADES NACIONALES





Cable Centronics

La impresora no siempre es un capricho, especialmente si de aplicaciones "profesionales" se trata. Con el QL lo más sencillo es utilizar una impresora Serie, pero también puede utilizarse las Centronics, siempre que disponga del interface adecuado. **Serma** comercializa el QL-Centronics con software incorporado, al precio de 12.500 pesetas.

Y más cables: el del joystick. Por el momento todos hablan de aplicaciones, pero ya existen diversos juegos que sin duda agradecerán el complemento. Comercializados igualmente por **Serma**, el precio es de 1.600 pesetas.

QL-Ventas

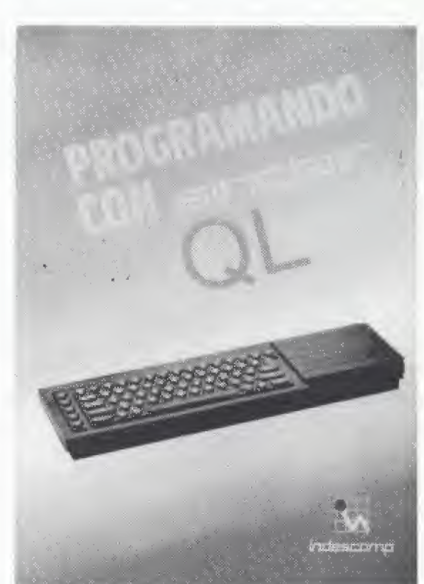
Las ventas del nuevo ordenador no parecen haber alcanzado las cotas previstas en el Reino Unido. Para el público inglés no hay todavía demasiado software que aconseje su compra. Conscientes de ello, Sinclair va a lanzar una campaña publicitaria cuyos costes se elevan a 500.000 libras (poca cosa, sólo 100 millones de pesetas). Y una suma adicional de 250.000 libras para las publicaciones especializadas. (¡A nosotros no nos llega nada!)

El resultado de esta inversión publicitaria será la venta de al menos 200.000 QL para este año en el Reino Unido. O al menos esos son los planes de Sinclair.

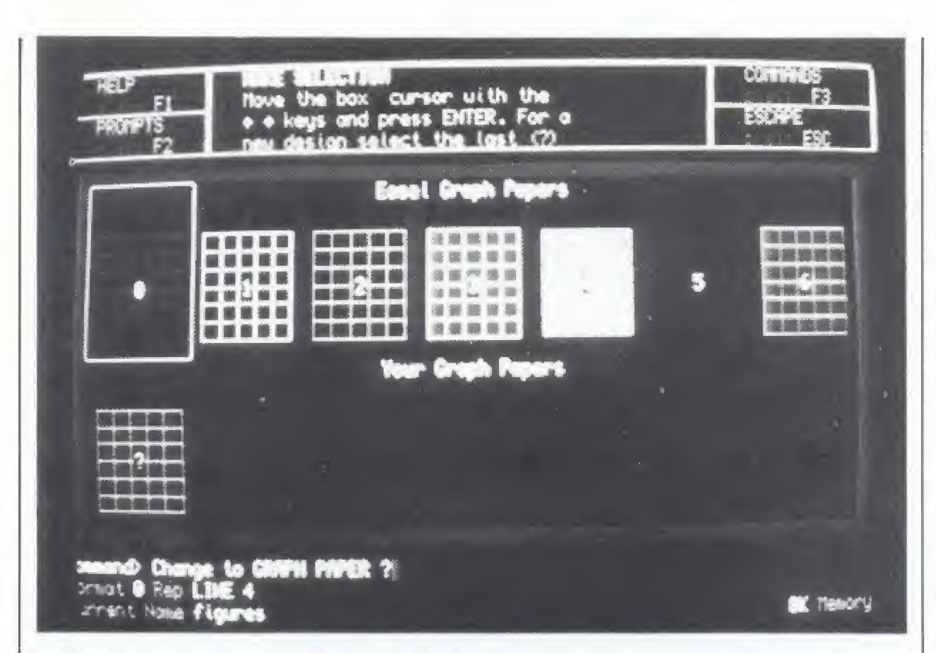
Curiosamente, los analistas predicen mayores ventas del QL en España que en su propio país. Habrá que esperar los resultados.

Libros

La primavera ya ha tenido en el Reino Unido el "florecimiento" de una abundante literatura sobre el QL. En España ya son cuatro los libros sobre los que tenemos noticias. A los dos que comentábamos en el número anterior se unen ahora "Programando con Sinclair QL" y "QL SuperBasic", editados ambos por **Indescomp.** Texto introductorio en el primer caso, y curso avanzado sobre el SuperBasic, en el segundo. El precio de cada libro es de 1.950 pesetas.







EASEL: Lo mejor de Psion

Comenzamos con este artículo una serie dedicada al *software* de **Psion** incluido en el precio del QL. En tanto estén disponibles las versiones en castellano de los programas, comenzaremos por la descripción de Easel (versión 1.01). Las únicas diferencias con la versión en castellano residirán en la traducción de los nombres de comandos, así como de una mayor velocidad de trabajo y capacidad de datos.

Easel es un programa de gráficos profesionales. No nos servirá, por tanto, para realizar dibujo "artístico"; su misión es realzar la presentación de nuestros resultados mediante gráficos. Muchas personas lo encontrarán de escasa utilidad, ya que este tipo de utilidades no resuelven una necesidad: la crean. Las personas que realizan informes o trabajos en los que la presentación de tablas numéricas resulte importante, descubriran que al poco tiempo de usarlo les resultará difícil prescindir de sus posibilidades. Las "víctimas" de sus documentos le agradecerán la mejora en la legibilidad de sus tablas.

Una de sus mejores características es la facilidad de uso. Su adaptación "en tiempo real" a todas las órdenes que recibe por el teclado, hace su manejo muy sencillo, ya que el efecto de cada acción es visible en pantalla inmediatamente. Como los demás programas de **Psion**, la tecla F1 nos envía al menú de Ayuda (Help), donde se resumen las instrucciones de una manera esquemática.

Las teclas F2 y F3 son también estándar en los cuatro programas. La primera sirve para indicar, en la parte superior de la pantalla, un resumen de las instrucciones de manejo. Cuando esta ventana de "status" no está presente disponemos de más espacio para nuestros datos. La tecla F3 nos da acceso al modo "comando", que nos permite teclear órdenes. En modo de inserción podemos teclear texto, datos o fórmulas. La tecla F4 sirve para borrar un valor numérico, una etiqueta o un texto. F5 abre una celda para insertar un nuevo valor a la derecha de la posición del cursor. El movimiento del cursor se realiza mediante la tecla TABULATE o bien de forma automática tras la introducción de un valor. SHIF TAB mueve el cursor hacia la izquierda.

Desde el momento en que EASEL acaba de cargar en memoria, el programa espera un valor numérico que ocupará la primera celda (etiquetadas por defecto con los nombres de los meses). Si respondemos con un número, habremos introducido el valor en la primera posición del gráfico. Si tecleamos unas comillas, el programa esperará un texto. Después de pulsar ENTER se nos da opción a mover el texto por la pantalla, pudiendo posicionarlo a voluntad.

Se pueden cambiar las etiquetas de las celdas a voluntad, bastando para ello el comando Edit Labels. Podemos borrar con F4, introducir texto nuevo y, si pulsamos TAB, pasaremos a editar la etiqueta siguiente. Si, en cambio, pulsamos ENTER, saldremos del modo edición.

La representación por defecto se realiza en diagrama de barras, pero tanto la forma de las barras como el fondo pueden ser elegidos a voluntad. Si ninguna de las opciones nos satisface podemos incluso diseñar estos elementos a nuestro gusto. A todas las opciones del programa se accede mediante menús, lo que facilita su uso por personas sin conocimientos informáticos. Los ocho formatos predefinidos incluyen seis diferentes diagramas de barras, tres en sentido vertical y tres horizontales: un diagrama de líneas y una representación tipo tarta. Sin embargo, las definiciones de barras y líneas se pueden cambiar una por otra y mezclarse en el mismo dibujo. También es posible representar varias figuras simultáneamente. El formato tarta resulta mucho menos flexible: sólo se puede representar una figura, y en un formato único.

Si queremos cambiar la representación de una figura (y estamos trabajando con varias) basta teclear F3, Olddata y el nombre de la figura. Despues F3, Change y elegir si cambia-

EASEL

mos barra, línea, papel, etc. En cada caso se nos pide un número; si tecleamos ENTER, se nos presentará una pantalla con las diferentes opciones. Siempre la última opción es un "?" Si elegimos ésta, el programa nos presenta un menú para que diseñemos a nuestro gusto.

EASEL es un programa flexible, pero cuando adquiere su máxima potencia es cuando se utiliza en conexión con ABACUS. Una hoja de cálculo es un útil muy potente para la manipulación de datos numéricos, siendo la "falta de vida" de la presentación tabular su principal inconveniente. Si una vez procesados los datos se crea un fichero tipo Export desde ABACUS, su importación a EASEL limita el trabajo necesario para la presentación visual a la elección del formato.

La obtención de copias en papel es uno de los principales problemas de EASEL. Por el momento el programa funciona con impresoras serie compatibles Epson, por lo que, si se puede elegir, una impresora con estas características nos solucionará la papeleta. Si ya disponíamos de una impresora y no es de ese tipo, el problema se complica, no existiendo por el momento una solución satisfactoria.

Tampoco existe otra posibilidad, si queremos copias en color, que realizar una fotografía de la pantalla, ya que tampoco existe ningún programa de copia de pantalla para impresoras en color. Esperamos que, bien **Psion,** bien otras empresas, se apresuren a resolver este problema.

Existen programas que entran por los ojos, que nadie se puede resistir a utilizar. Si queremos demostrar las posibilidades gráficas del QL, éste es nuestro programa. Salvando las reducidas posibilidades de copia en papel, EASEL merece un diez.

Como construir un gráfico

Como un ejemplo del manejo de EASEL, supongamos que un usuario quiere presentar en forma gráfica la pérdida de poder adquisitivo de tres distintas categorías laborales en su empresa. Para ello serviría, por ejemplo, una gráfica como la figura 1, donde se presenta el Indice de Precios al Consumo (IPC) junto a la variación de los sueldos de las tres categorías para el período considerado. Para facilitar la comparación, se cambia la escala de cada sueldo, de manera que los tres sean iguales al IPC de 1976, primer año que coexistieron las tres categorías.

Para comenzar, se teclea F3, N *(Newdata)* y luego CAT1: primera categoría para la que introduciremos los sueldos. A continuación, E *(Edit)* L *(Labels).* Esto nos deja en situación de modificar las etiquetas. El proceso sería: F4 (borra la antigua), F1 (nueva), TAB (pasar a la siguiente), etc., hasta tener las celdas necesarias para nuestros datos. Volvemos a la primera celda (mediante SHIFT TAB) y comenzamos a teclear el sueldo bruto anual de cada año. Una vez acabada

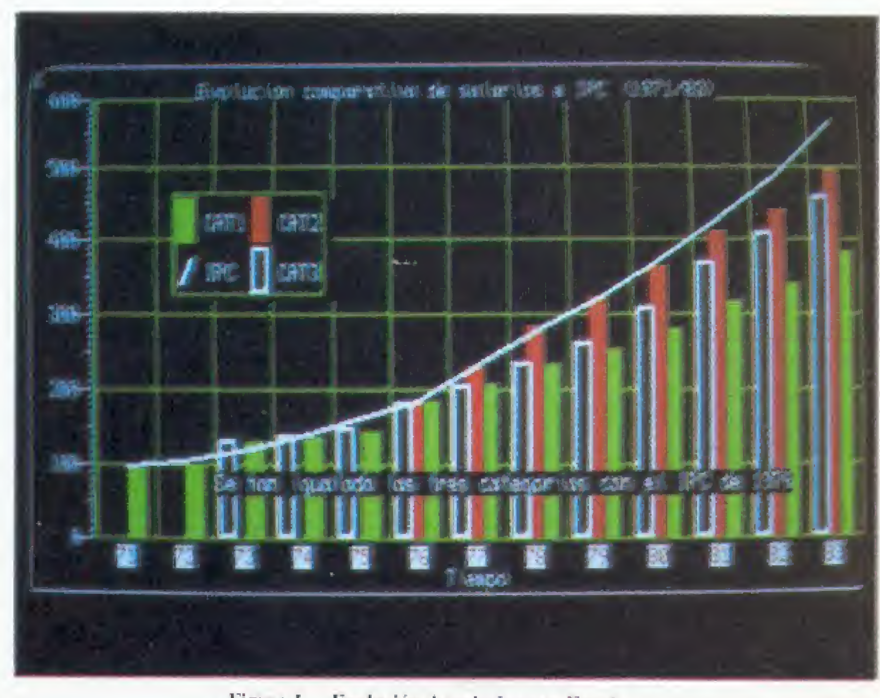


Figura 1. Evolución de salarios con Easel.

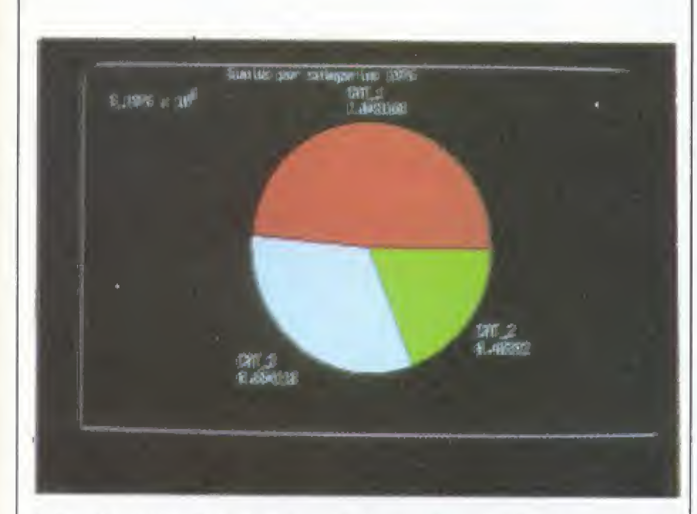


Figura 2. Ejemplo de "tarta". Sueldo por categorías 1976.

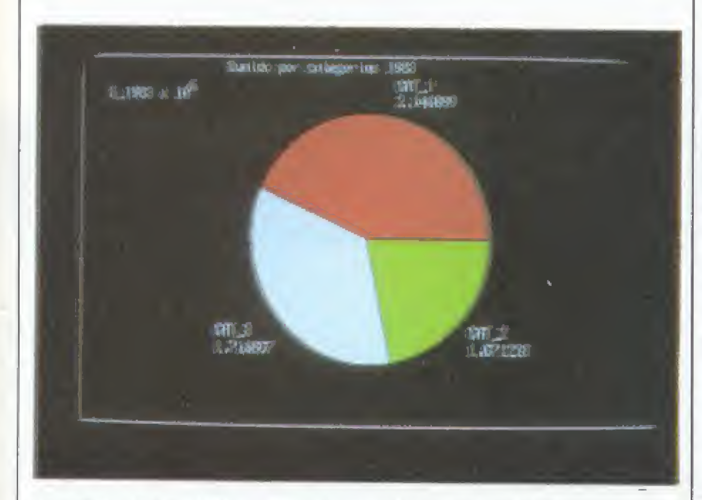


Figura 3. Ejemplo de "tarta". Sueldo por categorías 1983.

la serie se introduce la fórmula CAT1 = CAT1 / 1403168, que corresponde a hacer el sueldo de 1976 igual a 1, y luego CAT1 = CAT1 * 100, lo que iguala el sueldo de 1976 al coste de la vida acumulado (1976 = 100).

Un proceso análogo se realiza con las restantes categorías profesionales y finalmente se añade la gráfica del coste de la vida. Después se teclea C *(Change)*, L *(Line)* y se elige el tipo de línea deseado para esta gráfica. El comando V *(View)* nos permite visualizar el resultado, bastando editar el título y los nombres de los ejes para disponer de una gráfica de presentación impecable.

Mediante este pequeño ejemplo hemos cubierto parte de las posibilidades de este programa. Se puede, además, resaltar el cierre del abanico salarial en el período considerado mediante otras dos representaciones, de tipo tarta, con los salarios de las tres categorías en 1976 y en 1983. Si, tras salvar a *microdrive* los datos anteriores, ejecutamos el comando Zap, tendremos nuevamente la pantalla limpia. Editando las etiquetas como CAT1, CAT2 y CAT3, e introduciendo en las celdas los valores del salario en el año 1976, bastará hacer *View*, respondiendo ENTER y 7 a los requerimientos del programa. Con esto quedará realizada la figura 2. La figura 3 es análoga, pero sus datos se refieren al año 1983. Mucho más presentable que una masa ilegible de números, que nadie, a menos que sea imprescindible, querrá mirar.

La figura 4 responde a otro tipo de aplicación: la representación de funciones. Un campo propio de estudiantes y profesores, ingenieros, etc. En este caso hemos representado una distribución de gauss y su derivada (no a escala). Para realizarla se utilizaron 67 valores. Tras introducir las etiquetas para 67 puntos, se teclea la fórmula:

```
gauss = exp
(-(cell-30)^2/10)
```

y el dibujo aparece ante nuestros ojos. Añadiendo:

```
deriv = -2*(cell-
30)/10*gauss)
```

creamos la representación de la derivada. Con ajustar la escala y los formatos, hecho.

Como puede verse, crear un gráfico con Easel resulta tan rápido como sencillo. Sin duda, una de las mejores armas de Sinclair para la promoción y venta del QL. En el próximo número veremos cómo aprovecharle mejor mediante la utilización del Abacus.

Figura 4. Gráficos para representar todo tipo de funciones.

SONIDO CON QL

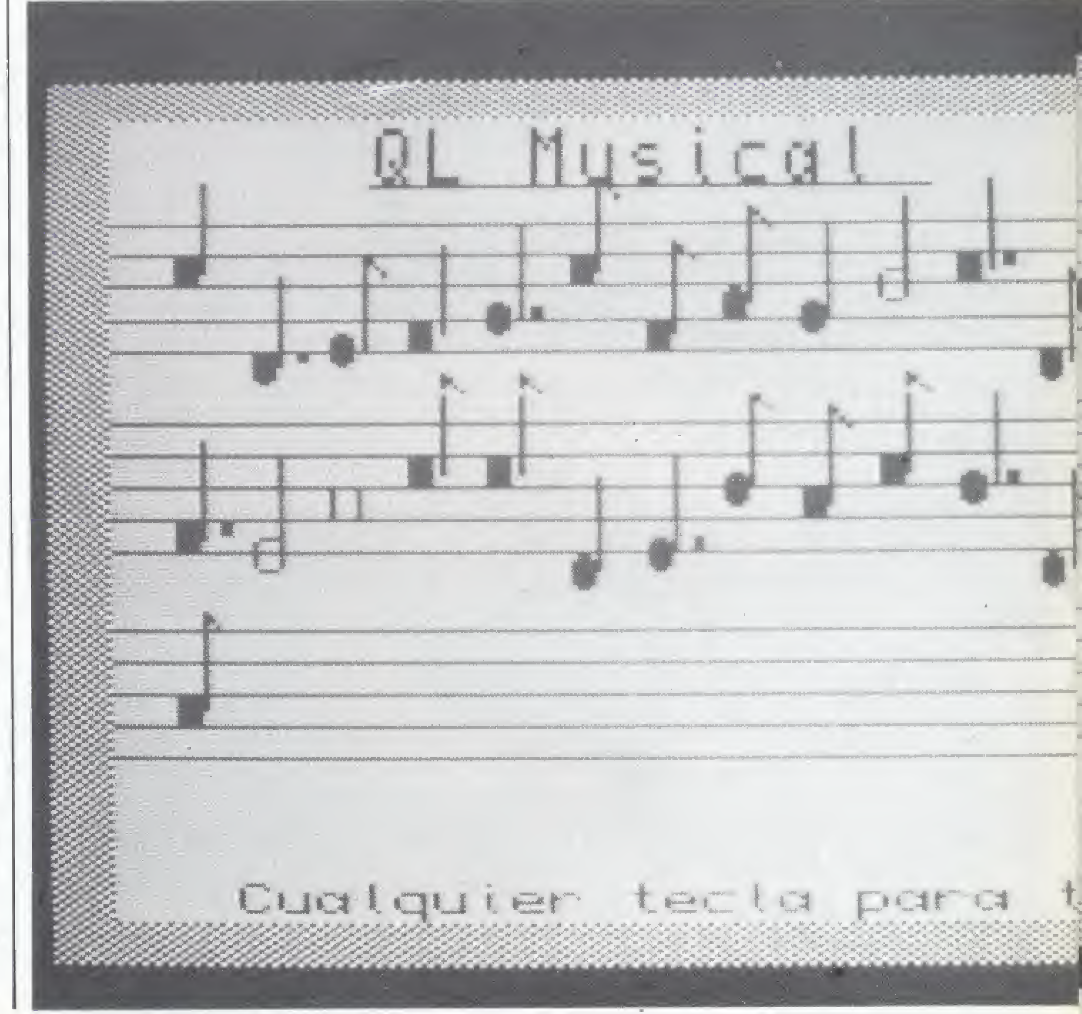
El QL nos proporciona un generador de sonido que, aunque rudimentario, supera apreciablemente las posibilidades musicales del Spectrum, ya que el uso del segundo procesador permite emitir sonido mientras se ejecuta un programa. El control del altavoz desde el BASIC se realiza mediante un solo comando (BEEP), que puede tener hasta 8 parámetros. El efecto de una pequeña variación en sus valores resulta a veces imprevisible, por lo que el propio manual nos recomienda la experimentación como mejor método para descubrir qué se puede hacer con el altavoz del QL.

El BEEP se puede utilizar como en el Spectrum, aunque los valores de duración y tono no muestran una equivalencia. Donde las cosas empiezan a cambiar es con la posibilidad de añadirle un segundo tono a la lista de parámetros. En ese caso la frecuencia del sonido oscilará entre los dos tonos, tardando un tiempo proporcional a los dos parámetros siguientes, grad-x y grad-y. Los tres parámetros siguientes resultan aún más esotéricos. En un próximo número intentaremos aclarar su función.

Si la duración especificada es cero, el ordenador seguirá sonando para siempre o hasta que ejecutemos BEEP sin parámetros, lo que para el altavoz. Para permitirnos sincronizar la ejecución de programas existe una función (BEEPING) que devuelve uno si el QL está emitiendo sonido.

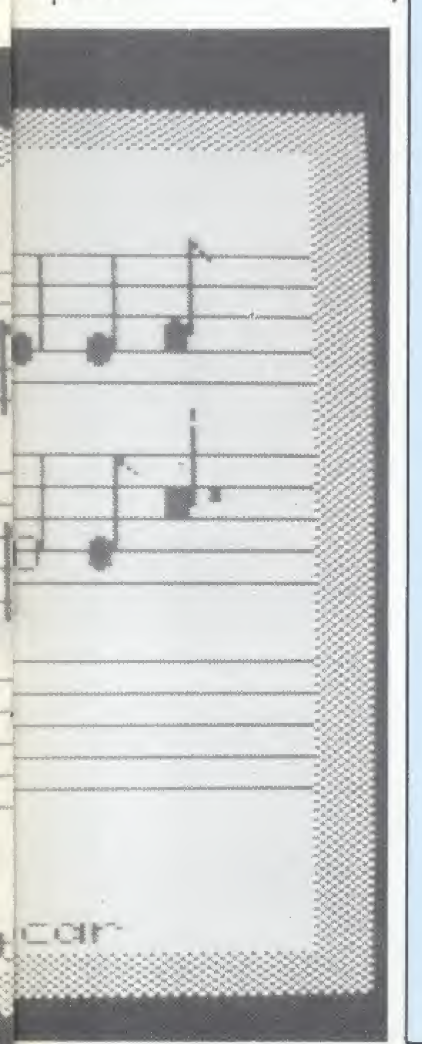
Incluimos un pequeño programa que demuestra las posibilidades sonoras del QL, se trata de aprender solfeo con ayuda del QL. El programa pedirá el tono y la duración de cada nota, hasta 45; las dibujará en un pentagrama y, al finalizar, interpretará la melodía.

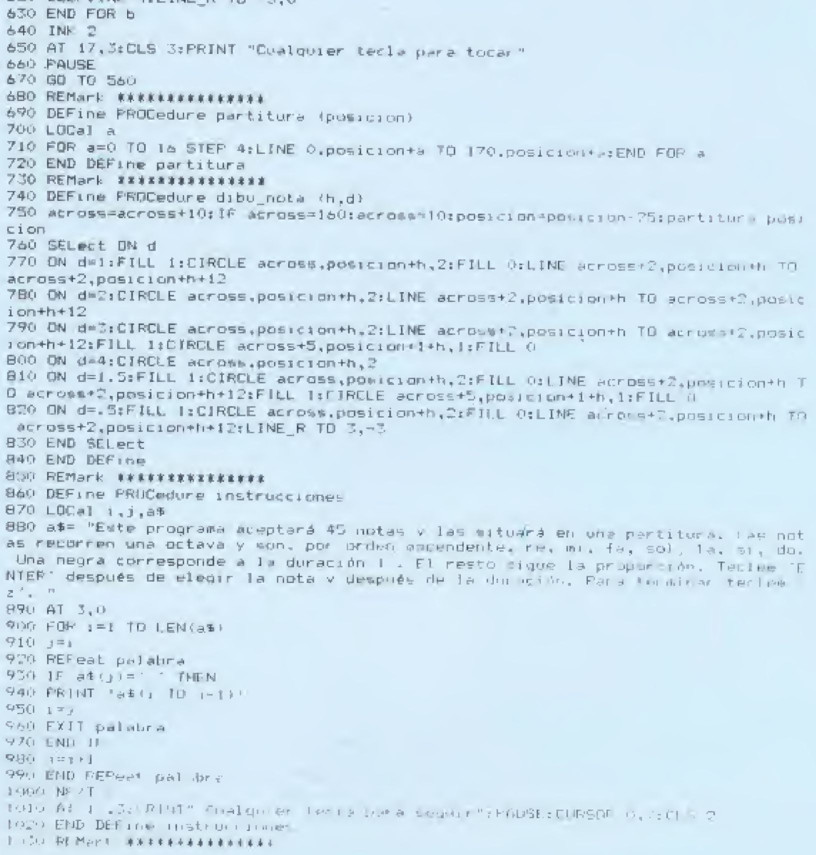
El tono de cada nota se introduce tecleando el nombre: do, re, mi..., aunque en la partitura la nota más baja es "re" y la más alta el "do" de la octava siguiente. En cuanto a la duración, 1 corresponde a una negra, 0,5 a una corchea, 1,5 a una negra con puntillo, 2 a una blanca, 3 a una blanca con puntillo y cuatro a una redonda. No se admiten otros



valores. El programa temporiza las notas mediante PAUSE, por lo que no debemos mantener ninguna tecla pulsada durante la interpretación.

Un programa sencillo que nos permitirá aprender solfeo a la vez que perfeccionamos nuestra programación en SuperBasic. Para que luego digan que el QL no sabe hacer música.





```
100 REMark ***************
110 REMark QL Musical
120 REMark ***************
130 REMark (c) Diciembre 1984, James Lucy
140 MODE 8:WINDOW 448,200,32,16:BORDER 10,0,7:PAPER 4:INK 0
150 CLS:CSIZE 3,1:UNDER 1:AT 0,5:PRINT "QL Musical ":UNDER 0:CSIZE 2,0:INK 2
160 PRINT:PRINT "¿ Quieres Instrucciones ? (s/n)"
170 i$=INKEY$(-1):IF i$="s" OR i$="S":instrucciones:ELSE CLS 3
180 DIM nota$(50,3):DIM duracion (50)
190 DIM pitch(50):DIM longitud(50)
200 a=0:posicion=70:across=0
210 partitura posicion
220 REPeat cargador
230 a=a+1
240 IF a=46:a=45:EXIT cargador
250 AT 17,5:PRINT"('ENTER' entre las dos)"
260 AT 16,5:INPUT"nota, duración?":nota$(a),a$:CLS 3:CLS 2
270 duracion(a)='0'&a$
280 IF nota$(a) INSTR "doremifasolasiz"
290 ELSE GO TO 260
300 END IF
310 IF nota$(a)="do" :pitch(a)=11:h=10
320 IF nota$(a)="re" :pitch(a)=28:h=-2
330 IF nota$(a)="mi" :pitch(a)=24:h=0
340 IF nota$(a)="fa" :pitch(a)=22:h=2
350 IF nota$(a)="sol" :pitch(a)=19:h=4
360 IF nota$(a)="la" :pitch(a)=15:h=6
370 IF nota$(a)="si" :pitch(a)=12:h=8
380 IF nota$(a)="z" :a=a-1:EXIT cargador
390 d=duracion(a)
400 SELect ON d
410      =.5:longitud(a)=6.25
420      =1:longitud(a)=12.5
430      =1.5:longitud(a)=18.75
440      =2:longitud(a)=25
450      =3:longitud(a)=37.5
460      =4:longitud(a)=50
470      =REMAINDER :GO TO 260
480 END SELect
490 dibu_nota h,d
500 BEEP-1,pitch (a)
510 PAUSE longitud(a)
520 BEEP
530 END REPeat cargador
540 AT 17,5:CLS 3:PRINT "Cualquier tecla para tocar"
550 PAUSE
560 across=0:posicion=70
570 FOR b=1 TO a
580 across=across+10:IF across=160:across=10:posicion=posicion-25
590 INK 7:LINE across,posicion-5 TO across+2,posicion-5
600 BEEP -1,pitch(b)
610 PAUSE longitud(b)
620 BEEP:INK 4:LINE_R TO -3,0
630 END FOR b
640 INK 2
650 AT 17,3:CLS 3:PRINT "Cualquier tecla para tocar"
660 PAUSE
670 GO TO 560
680 REMark ***************
690 DEFine PROCedure partitura (posicion)
700 LOCal a
710 FOR a=0 TO 16 STEP 4:LINE 0,posicion+a TO 170,posicion+a:END FOR a
720 END DEFine partitura
730 REMark ***************
740 DEFine PROCedure dibu_nota (h,d)
750 across=across+10:IF across=160:across=10:posicion=posicion-75:partitura posi
cion
760 SELect ON d
770 ON d=1:FILL 1:CIRCLE across,posicion+h,2:FILL 0:LINE across+2,posicion+h TO
across+2,posicion+h+12
780 ON d=2:CIRCLE across,posicion+h,2:LINE across+2,posicion+h TO across+2,posic
ion+h+12
790 ON d=3:CIRCLE across,posicion+h,2:LINE across+2,posicion+h TO across+2,posic
ion+h+12:FILL 1:CIRCLE across+5,posicion+1+h,1:FILL 0
800 ON d=4:CIRCLE across,posicion+h,2
810 ON d=1.5:FILL 1:CIRCLE across,posicion+h,2:FILL 0:LINE across+2,posicion+h T
O across+2,posicion+h+12:FILL 1:CIRCLE across+5,posicion+1+h,1:FILL 0
820 ON d=.5:FILL 1:CIRCLE across,posicion+h,2:FILL 0:LINE across+2,posicion+h TO
 across+2,posicion+h+12:LINE_R TO 3,-3
830 END SELect
840 END DEFine
850 REMark ***************
860 DEFine PROCedure instrucciones
870 LOCal i,j,a$
880 a$= "Este programa aceptará 45 notas y las situará en una partitura. Las not
as recorren una octava y son, por orden ascendente, re, mi, fa, sol, la, si, do.
 Una negra corresponde a la duración 1 . El resto sigue la proporción. Teclee 'E
NTER' después de elegir la nota y después de la duración. Para terminar teclee '
z'. "
890 AT 3,0
900 FOR i=1 TO LEN(a$)
910 j=i
920 REPeat palabra
930 IF a$(j)=' ' THEN
940 PRINT 'a$(i TO j-1)'
950 i=j
960 EXIT palabra
970 END IF
980 j=j+1
990 END REPeat palabra
1000 NEXT
1010 AT 1,3:PRINT"Cualquier tecla para seguir":PAUSE:CURSOR 0,0:CLS 2
1020 END DEFine instrucciones
1030 REMark ***************
```

Caracteres españoles en el QL.

Se han usado caracteres españoles en los mensajes e instrucciones de este programa. Para obtenerlos hay que teclear:

ñ	CNTRL	SHIFT	0
Ñ	CNTRL	SHIFT	1
á	CNTRL		,
é	CNTRL	SHIFT	3
í	CNTRL		3
ó	CNTRL		6
ú	CNTRL		9
ü	CNTRL		.
¡	CNTRL	SHIFT	S
¿	CNTRL	SHIFT	T


ESPECIFICAMENTE PARA QL
Mod. 1431 MZ 4
P.V.P. 74.500 Ptas.
INMEDIATAMENTE DISPONIBLE
ESPECIFICACIONES
MODELO:
CUB 1451/DQ3
14" QL MONITOR
Entrada RGB-TTL
Resolución (PIXELS)
653 (H) × 585 (V)
DOT PITCH 0,43 mm
Bandwidth 18 MHz
Específicamente diseñado para el QL que aprovecha su facilidad única de 85 columnas.
Mod. 1451/DQ3
P.V.P. 96.000 Ptas.
MICROVITEC 653
cub
COLOUR DISPLAYS
DISTRIBUIDO EN EXCLUSIVA POR: MULTILOGIC. P.° de la Habana, 145. 28036-MADRID Tel. 458 74 75

(viene de la pág. 30)

envía el *byte* en el acumulador al *port* 14

LD B, O
LD C, 10
IN A, (C)

manda un pulso de sincronización *(strobe)* al *port* 10

LD B, 20 h
LD C, 10
IN A, (C)
cancela el *strobe*

En lo concerniente al *software*, este es un modo de hacerlo tan bueno como cualquier otro, la excentricidad está en el *hardware*.

Otros interfaces

Para otros *interfaces* paralelos habrá que olvidar la rutina PRTER y colocar en el vector PRINT (FE40h y FE41h) la dirección de tu propia rutina de impresión. La mayoría de los *interfaces* comerciales de tipo Centronics incluyen en su ROM un corto programa equivalente a nuestra subrutina PRTER.



Ejemplo de «copy» de pantalla.

Las impresoras

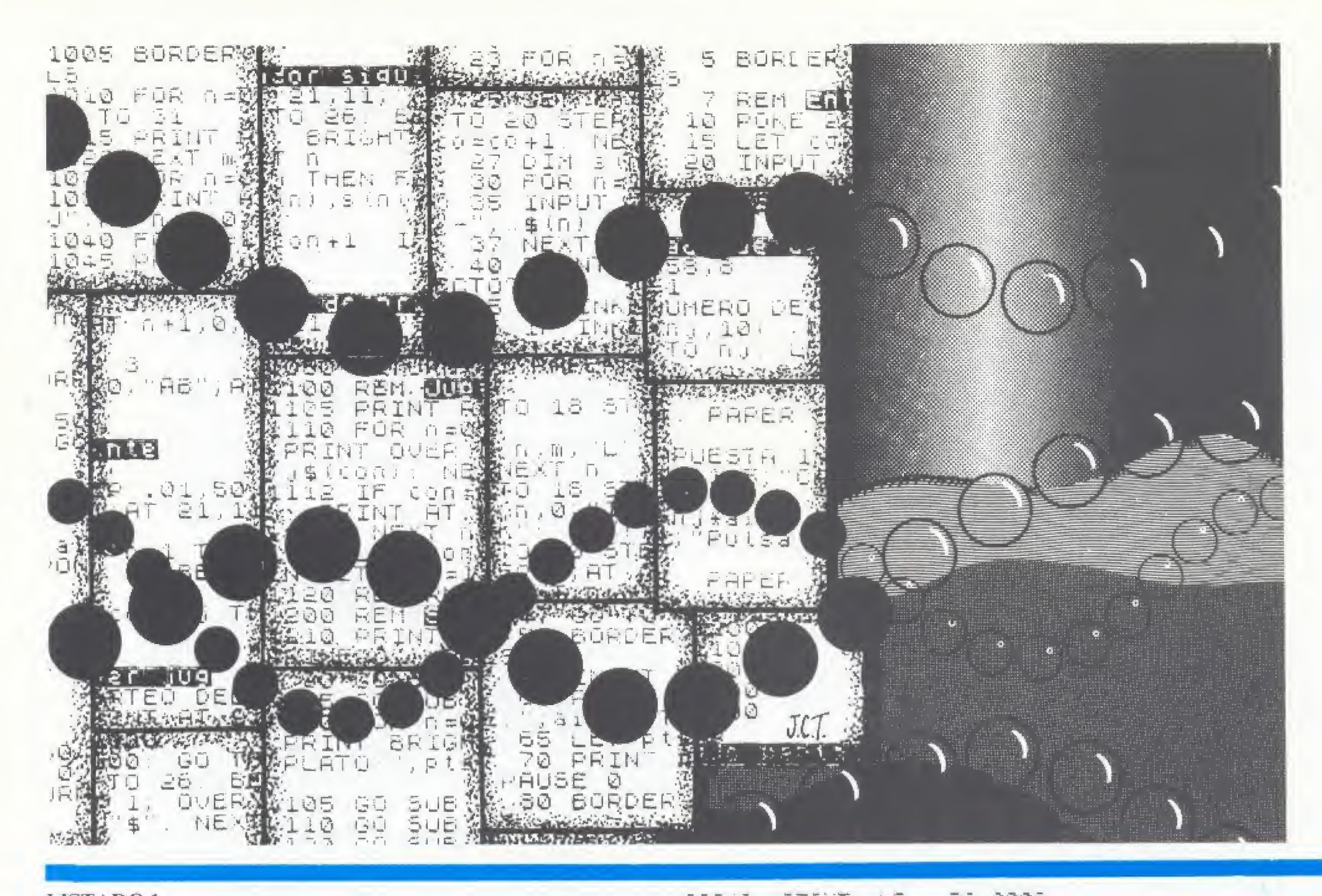
Imprimir en modo de alta resolución gráfica implica pasar a la impresora una serie de parámetros, que son entregados en el momento oportuno por tres subrutinas: EIGHT, NORMA y SINGL. Entre cada una de ellas se intercalan los suficientes NOPs para permitir hacer cambios.

Si tu impresora no ha sido ajustada para dar un *linefeed* automático (avance de la línea de impresión), deberás cambiar la instrucción NOP en FE3Dh por un CALL PRINT.

Para copiar la pantalla con la impresora, hacer un RANDOMIZE USR 64976. (Sin olvidar que el código máquina tiene que estar por encima del RAMTOP.) Como consecuencia de la forma en que trabaja el Spectrum, las dos últimas líneas quedarán en blanco. Esto se puede evitar llamando a la rutina desde dentro de un programa.

La creciente producción de aplicaciones que realizan gráficos hace que este corto programa pueda resultar muy útil a los poseedores del *hardware* necesario.

```
FE2F C1        0400        POP  BC
FE30 C9        0410        RET
FE31 00        0420        DEFB 0,0,0,0,0
     00 00 00 00
FE36 3E0D      0430 CR_LF  LD   A,0DH
FE38 CD3FFE    0440        CALL PRINT
FE3B 3E0A      0450        LD   A,0AH
FE3D 00        0460        NOP
FE3E C9        0470        RET
FE3F C342FE    0480 PRINT  JP   PRTER
FE42 C5        0490 PRTER  PUSH BC
FE43 F5        0500        PUSH AF
FE44 F5        0510        PUSH AF
FE45 010200    0520        LD   BC,2
FE48 ED78      0530 RDY    IN   A,(C)
FE4A CB6F      0540        BIT  5,A
FE4C 28FA      0550        JR   Z,RDY
FE4E F1        0560        POP  AF
FE4F 47        0570        LD   B,A
FE50 0E0E      0580        LD   C,14
FE52 ED78      0590        IN   A,(C)
FE54 0600      0600        LD   B,0
FE56 0E0A      0610        LD   C,10
FE58 ED78      0620        IN   A,(C)
FE5A C5        0630        PUSH BC
FE5B C1        0640        POP  BC
FE5C 0620      0650        LD   B,20H
FE5E 0E0A      0660        LD   C,10
FE60 ED78      0670        IN   A,(C)
FE62 F1        0680        POP  AF
FE63 C1        0690        POP  BC
FE64 C9        0700        RET
FE65 00        0710        DEFB 0,0,0,0,0
     00 00 00 00
FE6A CDD3FD    0720 SING   CALL EIGHT
FE6D 0600      0730        LD   B,0
FE6F 0E00      0740 COOR1  LD   C,0
FE71 DD21005B  0750        LD   IX,BUFFE
FE75 79        0760 COOR2  LD   A,C
FE76 CB3F      0770        SRL  A
FE78 6F        0780        LD   L,A
FE79 78        0790        LD   A,B
FE7A E630      0800        AND  30H
FE7C 0F        0810        RRCA
FE7D 67        0820        LD   H,A
FE7E 78        0830        LD   A,B
FE7F E60E      0840        AND  0EH
FE81 07        0850        RLCA
FE82 07        0860        RLCA
FE83 07        0870        RLCA
FE84 07        0880        RLCA
FE85 B5        0890        OR   L
FE86 6F        0900        LD   L,A
FE87 78        0910        LD   A,B
FE88 E601      0920        AND  1
FE8A 07        0930        RLCA
FE8B 07        0940        RLCA
FE8C B4        0950        OR   H
FE8D F640      0960        OR   40H
FE8F 67        0970        LD   H,A
FE90 C5        0980        PUSH BC
FE91 0608      0990        LD   B,8
FE93 E5        1000 ROTAT  PUSH HL
FE94 1600      1010        LD   D,0
FE96 3E08      1020        LD   A,8
FE98 4E        1030 F2     LD   C,(HL)
FE99 58        1040        LD   E,B
FE9A CB39      1050 F3     SRL  C
FE9C 1D        1060        DEC  E
FE9D 20FB      1070        JR   NZ,F3
FE9F CB12      1080        RL   D
FEA1 24        1090        INC  H
FEA2 3D        1100        DEC  A
FEA3 20F3      1110        JR   NZ,F2
FEA5 7A        1120        LD   A,D
FEA6 DD7700    1130        LD   (IX+0),A
FEA9 DD23      1140        INC  IX
FEAB E1        1150        POP  HL
FEAC 10E5      1160        DJNZ ROTAT
FEAE C1        1170        POP  BC
FEAF 0C        1180        INC  C
FEB0 0C        1190        INC  C
FEB1 79        1200        LD   A,C
FEB2 FE40      1210        CP   64
FEB4 38BF      1220        JR   C,COOR2
FEB6 CD06FE    1230        CALL SINGL
FEB9 04        1240        INC  B
FEBA 04        1250        INC  B
FEBB 78        1260        LD   A,B
FEBC FE30      1270        CP   48
FEBE 38AF      1280        JR   C,COOR1
FEC0 CDF3FD    1290        CALL NORMA
FEC3 C9        1300        RET
               1310        END
```



LISTADO 1

```
        ;       ****************
        ;       **   PIXEL    **
        ;       ****************

        ;       Programa enviado  por
        ;       Alvaro Mateos Herrera

MOVER   LD      HL,(STKEND)
        PUSH    HL
        EX      DE,HL
        LD      HL,(STKBOT)
        LD      BC,000FH; 5*3 = 0FH
        ADD     HL,BC
        SBC     HL,DE
        JR      Z,FCODE
        RST     08
        DEFW    19H

FCODE   CALL    POPA
        CP      80H
        JR      C,ASCII; Car ASCII
        LD      B,A
        SUB     90H
        JR      C,BLOCK; Car GRAFICO
        LD      DE,(UDG);     UDG
        JR      INDEX

ASCII   LD      DE,(CHARS)
INDEX   LD      H,00
        LD      L,A
        ADD     HL,HL
        ADD     HL,HL
        ADD     HL,HL
        ADD     HL,DE
        JR      GFONT

BLOCK   CALL    SYNTB
        LD      HL,MEMBOT
GFONT   LD      BC,0008
        ADD     HL,BC
        PUSH    HL
        POP     IX
        CALL    POPA
        LD      H,A;    Coord Y
        PUSH    HL
        CALL    POPA
        POP     HL
        LD      L,A;    Coord X
        LD      B,H
        LD      C,L
        CALL    PIXEL
        PUSH    HL
        PUSH    HL
SIGUE   LD      B,A
        LD      A,8
        SUB     B
        LD      B,A
        LD      C,8;          Contador
LOOP    LD      A,B
        CP      8
        LD      A,(IX+0)
        JR      Z,STORE
        EX      DE,HL
        LD      H,0
        LD      L,A
        PUSH    BC
SHIFT   ADD     HL,HL
        DJNZ    SHIFT
        POP     BC
        EX      DE,HL
        LD      A,D
        XOR     (HL)
        LD      (HL),A
        INC     HL
        LD      A,E
        XOR     (HL)
        LD      (HL),A
        DEC     HL
        JR      CONT
```

Desplazamiento

Pixel a Pixel

EN el número 2 de TODOSPECTRUM se invitaba a los lectores a comentar un programa del citado ejemplar. Y hubo quienes, no contentos con lo que se ofrecía, decidieron modificarlo para dotarlo de una mayor potencia. Tal fue el caso de Alvaro Mateos que nos envió dos rutinas inspiradas en aquélla.

La primera de ellas (Listado 1) permite el control de SPRITES con sus atributos, innovación no presente en nuestro programa. La hemos probado y nos ha parecido muy interesante. Ahora compartimos el hallazgo con nuestros lectores, y aprovechamos para guiarles por el mundo siempre apasionante de la programación en ensamblador.

Mencionaremos rápidamente el modo de acceder a la rutina. Será mediante la instrucción.

INK u: PAPER v: RANDOMIZE x AND y = c — USR dir, donde

u y *v* son números entre 0 y 7, los que deseemos para el gráfico a imprimir.

x e *y* son coordenadas del extremo superior derecho del gráfico a imprimir.

c es el CODE del gráfico: ASCII, carácter gráfico o UDG.

dir es la dirección de la subrutina (por cierto, es relocalizable).

Si desea conocer más detalles les remitimos al citado ejemplar de la revista.

```
00081
00082  STORE   XOR   (HL)
00083          LD    (HL),A
00084  CONT    DEC   H
00085          LD    A,H
00086          CP    3FH
00087          JR    Z,PALI
00088          CP    47H
00089          JR    Z,PALI
00090          CP    4FH
00091          JR    Z,PALI
00092          JR    NPAL
00093
00094  PALI    ADD   A,08
00095          LD    H,A
00096          LD    A,L
00097          SUB   20H
00098          LD    L,A
00099          JR    NC,NPAL
00100          LD    A,H
00101          SUB   08
00102          LD    H,A
00103  NPAL    DEC   IX
00104          DEC   C
00105          JR    NZ,LOOP
00106          LD    C,02
00107  BUCLE   LD    A,H
00108          AND   18H
00109          SRA   A
00110          SRA   A
00111          SRA   A
00112          ADD   A,58H
00113          LD    H,A
00114          LD    A,(ATTR)
00115          LD    (HL),A
00116          INC   HL
00117          LD    (HL),A
00118          POP   HL
00119          DEC   C
00120          JR    NZ,BUCLE
00121          POP   HL
00122          LD    (STKEND),HL
00123          RET
00124
00125
00126  STKEND  EQU   5C65H
00127  POPA    EQU   2DA2H
00128  MEMBOT  EQU   5C92H
00129  STKBOT  EQU   5C63H
00130  UDG     EQU   5C7BH
00131  CHARS   EQU   5C36H
00132  SYNTB   EQU   0B38H
00133  PIXEL   EQU   22AAH
00134  ATTR    EQU   5C8DH
```

Descripción del programa

Líneas 9-10: Guardaremos en el stack la dirección del puntero de pila del calculador, con objeto de devolver su valor inicial al volver al BASIC.

Líneas 11-16: Comprobaremos si la pila del calculador guarda los tres números correspondientes a las coordenadas y al código del carácter. Teniendo en cuenta que dichos números se almacenan en formato de punto flotante, y que cada uno de estos números ocupa 5 bits, sólo tendremos los citados tres números si el *stack* tiene una longitud de 15 *bytes.*

Si no ocurriera de este modo, detendríamos el programa dando un mensaje de error. Con este objeto llamamos a la rutina de dirección 8 (rutina en página cero, por tanto usaremos RST y no CALL). El siguiente *byte* guardará el código de error a generar.

Línea 20: La subrutina de ROM, FP-TO-BC (2DA2H), a la que hemos llamado POP-A, tiene la misión de almacenar en BC el último número de la pila del calculador. También se encargará de actualizar el puntero de pila.

Ya hemos comentado que dicho número está en formato de punto flotante (*FLOATING POINT,* FP). Consecuentemente solamente será posible esto en el caso de que su valor esté comprendido entre 0 y 65535. El Acumulador también guarda una copia del *byte* menos significativo. Como éste es el único que nos interesa, solamente haremos uso del registro A.

Línea 21-27: El código de carácter, almacenado en A, se analiza para ver a cuál de los siguientes grupos pertenece:

— Caracteres ASCII (entre 64 y 127).

— Caracteres gráficos (entre 128 y 143).

— Caracteres definidos por el usuario UDG (145-164).

No se realiza una comprobación muy exhaustiva de los posibles valores, por lo que cifras por debajo de 63 o por encima de 165 dará resultados impredictibles.

Las medidas a tomar en el caso de caracteres ASCII o UDG, son

dobles: Almacenar en DE las direcciones de comienzo de las tablas de caracteres. (En realidad se utilizan ligeros apaños que facilitan los cálculos). Líneas 29 y 26 respectivamente. Y, en segundo lugar, calculamos la dirección donde vamos a encontrar la información del gráfico que nos ocupa. Para ello multiplicaremos el valor del código (restando 144 en el caso de un UDG) por ocho. Esto se consigue duplicándolo tres veces. Por último sumaremos con DE, con lo que ya tendremos la dirección exacta. (Líneas 30-35).

En el caso de caracteres gráficos, los perfiles no se encuentran en memoria, pues son calculados por una rutina, que hemos llamado SYNTB. Dicha subrutina (OB38H), situada en ROM, deja en los 8 primeros *bytes* de MEMBOT (área auxiliar del calculador) las "rebanadas" del carácter gráfico cuyo código se almacena en el registro B.

Cargaremos en HL la dirección del primero de los *bytes*, como en otros casos.

Línea 40: En esta línea convergen las tres variantes. Antes de almacenar HL en IX, sumaremos 8 a este primer registro, con objeto de apuntar al último *byte* de la minitabla que define el gráfico. ¿Por qué? Pues porque vamos a dibujar el gráfico de abajo a arriba.

Se mejoró la rutina, incorporando el control de Sprites con sus atributos

Líneas 44-49: Este es el momento de recoger los datos de las coordenadas, almacenadas en la pila del calculador. Tras hacer algunos juegos de malabares con los registros, se almacenarán definitivamente en el par HL, "x" en L e "y" en H. Transferiremos su contenido a BC (líneas 50 y 51). Con este dato, la rutina PIXEL (de ROM, 22AAH) devolverá HL con la dirección del área de pantalla que corresponde a las coordenadas dadas. El contenido A hace referencia al bit correspondiente al pixel indicado. Exactamente A es x módulo 8. Restamos 8 de A (luego veremos porqué), y almacenamos el resultado en B (líneas 55-58).

Si inicialmente A hubiera estado a 0 (el pixel apuntaba al bit 7), no será necesario hacer rotación alguna. La comprobación se realizará en las líneas 60 y 61. Si así ocurriera, iríamos directamente a STORE.

Líneas 65-66: En otro caso, el caracter a imprimir estará a caballo entre dos *bytes*. Por tanto será necesario crear una máscara de 16 bits. Usaremos para ello el par HL. Almacenando la "rebanada" del caracter en L, y poniendo H a 0, rotaremos el registro tantas veces



¡APROVECHA AL MAXIMO TU SPECTRUM!

Ahora, a tu alcance, dos obras fundamentales para que podáis sacar todo el partido posible a vuestro ordenador.



Esta publicación está diseñada para guiar al nuevo usuario del ZX Spectrum desde el momento que el ordenador se conecta hasta conseguir una base suficiente de la programación BASIC.

Incluye temas como:

- Introducción al teclado.
- Instrumentos útiles para la programación.
- Uso de comandos fáciles.
- Como construir un programa.
- Técnicas de programación.
- Aplicaciones prácticas.

100 pags. - 750 PTAS.



Este libro, escrito en estilo ameno y práctico, está dirigido a todos aquellos usuarios que han dejado atrás la etapa de los juegos y necesitan adentrarse en el fabuloso mundo de la programación.

El temario incluye:

- Reglas y herramientas del BASIC.
- La técnica de los organigramas.
- Cómo planificar un programa.
- El mundo de las rutinas.
- Variables y cadenas.
- Funciones matemáticas usuales.

109 pags. - 750 PTAS.

CUPON DE PEDIDO

Recorta este cupón debidamente cumplimentado y envielo a INFODIS, S. A. C/ BRAVO MURILLO, 377-5.° A - 28020 MADRID

Si, envienme el(los) libro(s) que a continuación detallo al precio de 750 ptas. libro, más 100 ptas. en concepto de gastos de embalaje y envio.

El importe lo abonare: POR CHEQUE □ CONTRAREEMBOLSO □ CON TARJETA DE CREDITO (VISA □) (AMERICAN EXPRESS □) (INTERBANK □)

Número de mi tarjeta

TITULO ______________________

____

NOMBRE ______________________

CALLE ______________________

CIUDAD ______________ D. P. ______________

PROVINCIA ______________________

Firma

como haga falta. ¿Cuántas? Las que almacena el registro B tras las anteriores manipulaciones.

Líneas 67-70: Las rotaciones se harán a base de sumas (tengamos en cuenta que desplazar los bits a la izquierda supone doblar el contenido de un registro).

Líneas 71-79: La máscara construida en HL se pasará a DE y se superpondrá al contenido actual de la pantalla, mediante un XOR (equivalente a OVER 1).

Líneas 84-92: Ahora toca el turno de actualizar la dirección a donde se va a mandar el siguiente dato. Mientras no se cambia de renglón, la ley de variación será el decremento del registro h (esto es restar 256 a la dirección) —recordemos que estamos escribiendo el caracter de abajo a arriba—. En estos casos iremos directamente a NPAL.

Pero al hacer un cambio de renglón tenemos complicaciones. Si cambiar de línea supone cambiar de "área" (es del dominio público que la pantalla está dividida en tres bloques), el cambio anterior no es válido. Mientras que el *byte* alto de la dirección permanece inalterado, el bajo hay que decrementarlo en 20H. Si se trata de un cambio de renglón simple, será necesario sumar 7 al *byte* alto (8 en nuestro caso, pues ya le habíamos quitado 1) y restar 20H al bajo, como en el caso anterior (líneas 94-102).

La pregunta es ¿cómo sabremos que hay un cambios de renglón o de bloque?

El cambio de línea tiene lugar sólo en los casos en que H contiene los valores hexadecimales 3F, 47 o 4F. Ver líneas 86-90.

La segunda subrutina permite conocer los atributos correspondientes a un pixel dado

El cambio de renglón supone además, cambio de bloque si L está comprendido entre 00 y 1FH. Como en cualquier caso será necesario restar 20H a este registro: lo tenemos muy fácil. Realizamos la operación, y si el resultado tiene signo negativo (el *carry flag* está en 1), es porque cambiamos de bloque. En este caso debemos devolver a H su antiguo valor (líneas 100-120).

Líneas 103-104: Ya estamos preparados para pelearnos con el siguiente dato. Incrementamos el puntero que lo selecciona y disminuimos en una unidad el registro C, que hemos usado como contador de líneas. Incialmente se le ha dado el valor 8 en la línea 56. Cuando llegue a 0, será momento de ocuparnos de la actualización de los atributos.

Igualmente usaremos C como contador, aunque en este caso serán dos el número de iteraciones (línea 106).

Líneas 107-113: A partir de la dirección de la "rebanada" más alta del gráfico, calculamos la dirección de los atributos correspondientes a dichos puntos.

Líneas 114-117: Introducimos en esta dirección el valor correspondiente a los atributos actuales. Por regla general el gráfico ocupa dos bytes. Por eso también lo introducimos en el siguiente.

Repetimos el proceso para la "rebanada" más baja, con los datos anteriormente almacenados en el *stack*.

Por último, antes devolver al BASIC, dejaremos el puntero de la pila del calculador en su valor original (línea 121-122).

Alvaro Mateos también nos ha remitido otra subrutina que permite conocer los atributos correspondientes a un pixel dado (Listado 2). Para utilizarla se escribirá:

RANDOMIZE × AND y = USR dir

El valor de los atributos aparecerá en una de las variables del sistema no empleadas, la que se encuentra en la dirección 236814 (5C81H).

Con ello finalizamos —por el momento— el apasionante tema del "Deplazamiento pixel a pixel".

LISTADO 2

```
        ;       **************
        ;       **   ATTR   **
        ;       **************

        ;       Programa enviado  por
        ;       Alvaro Mateos Herrera

MOVER   LD   HL,(STKEND)
        PUSH HL
        EX   DE,HL
        LD   HL,(STKBOT)
        LD   BC,000AH; 5*2 = 0AH
        ADD  HL,BC
        SBC  HL,DE
        JR   Z,ACEPT
        RST  08
        DEFU 19H

ACEPT   CALL POPA
        LD   B,A;         Coord Y
        PUSH BC
        CALL POPA
        POP  BC
        LD   C,A;         Coord X
        CALL PIXEL
        LD   A,H
        AND  18H
        SRA  A
        SRA  A
        SRA  A
        ADD  A,58H
        LD   H,A
        LD   A,(HL)
        LD   (ATTRP),A
        POP  HL
        LD   (STKEND),HL
        RET


STKEND  EQU  5C65H
POPA    EQU  2DA2H
STKBOT  EQU  5C63H
PIXEL   EQU  22AAH
ATTRP   EQU  5C81H
```

Guía del comprador de Todospectrum



Pitufolandia

Erase una vez un viejo huraño y haraposo al que todos llamaban Gargamel... En realidad, en este juego no aparece este personaje, pero sí un enorme pitufo con preciosos tonos azulados que ha de recorrer incansablemente un laberinto, a fin de recoger el mayor número posible de pitufofresas, robadas por el malvado Gargamel.

El programa consta de tres listados, todos realizados en BASIC. Sin embargo, únicamente el último contiene el juego propiamente dicho. Los anteriores preparan diversas pantallas de carga, de gran calidad, y han supuesto mu-

LISTADO 1

```
   1 PAPER 0: POKE 23659,0
   2 REM
   3 POKE 23659,2
  10 GO SUB 180
  12 POKE 23659,0: POKE 23659,2
  20 GO SUB 35
  25 PRINT  INK 1;" PULSE PLAY": LOAD "2"
  26 REM
  28 STOP
  35 PAPER 0: CLS : BORDER 0
  36 POKE 23659,0: POKE 23659,2
  38 PAPER 0: BORDER 0: INK 7
  43 PAPER 0: BORDER 0: PRINT  INK 6;AT 1,1;"[block graphics]"; INK 3;AT 2,1;"[block graphics]"; INK 6;AT 3,1;"[block graphics]"
  44 PRINT  INK 6;AT 1,7;"[block graphics]"
  45 PRINT  INK 3;AT 2,7;"[block graphics]"
  46 PRINT  INK 6;AT 3,7;"[block graphics]"
  47 PRINT  INK 6;AT 1,14;"[block graphics]"
  48 PRINT  INK 3;AT 2,14;"[block graphics]"
  49 PRINT  INK 6;AT 3,14;"[block graphics]"
  50 PRINT  INK 3;AT 1,22;"[block graphics]"
  51 PRINT  INK 5;AT 2,22;"[block graphics]"
  52 PRINT  INK 3;AT 3,22;"[block graphics]"
  60 INK 7: PLOT 0,140: DRAW 255,0
  61 PLOT 0,142: DRAW 255,0: DRAW 0,1: DRAW -255,0
  62 PLOT 0,170: DRAW 255,0: DRAW 0,1: DRAW -255,0
  63 PLOT 0,173: DRAW 255,0
  64 PLOT 0,0: DRAW 0,175: DRAW 255,0: DRAW 0,-175: DRAW -255,0
  65 PLOT 0,110: DRAW 20,6: DRAW 4,3: DRAW 2,3
  66 DRAW 4,2: DRAW 12,3: DRAW 3,2: DRAW 5,11: DRAW 6,0
  67 DRAW -2,-12: DRAW 2,-3: DRAW 4,-1: DRAW 4,0: DRAW 2,3: DRAW 3,1
  68 DRAW 1,0: DRAW 4,1: DRAW 8,11: DRAW 4,0: DRAW -3,-11
  69 DRAW 2,-3: DRAW 4,-1: DRAW 2,0: DRAW 9,3
  70 DRAW 3,0: DRAW 5,4: DRAW 9,3
  71 DRAW 12,2: DRAW 5,-1
  72 DRAW 4,-1: DRAW 3,-9
  73 DRAW 0,-2: DRAW -2,-8: DRAW -4,-6
  74 DRAW 3,9: DRAW 0,2: DRAW -1,7
  75 DRAW -2,4: DRAW -6,-2
  76 DRAW -2,-4: DRAW 5,-6: DRAW 2,-5: DRAW ,-2: DRAW -3,-8: DRAW 0,7
  77 DRAW -2,7: DRAW -7,4: DRAW -2,0: DRAW -2,-12: DRAW -1,9
```




```
  78 DRAW -2,5: DRAW -12,-6: DRAW -12,-10
  79 DRAW -9,-2: DRAW -3,6: DRAW -3,0: DRAW -7,-2
  80 DRAW -20,-5: DRAW -2,-6: DRAW -4,-2
  81 DRAW -6,-2: DRAW -3,-7: DRAW -5,-2
  82 DRAW -28,-28
  84 PLOT 90,40: DRAW 3,12
  85 DRAW -2,3: DRAW -1,0: DRAW -2,-1: DRAW 2,6
  86 DRAW 4,3: DRAW 5,3: DRAW 30,6
  87 DRAW 5,2: DRAW 4,1
  88 DRAW 0,4: DRAW 2,0: DRAW 0,-4: DRAW -8,-44: DRAW 1,4: DRAW -44,4
  89 PLOT 90,55: DRAW 10,1: DRAW 1,-2: DRAW 2,3: DRAW 4,1: DRAW 2,-3: DRAW 2,1: DRAW 4,2
  90 DRAW 22,2
  91 PLOT 90,57: DRAW 10,1: DRAW 1,-2: DRAW 2,3: DRAW 4,1: DRAW 2,-3: DRAW 2,1: DRAW 4,2: DRAW 2,2
  92 PLOT 90,54: DRAW 10,1: DRAW 1,-2: DRAW 2,3: DRAW 4,1: DRAW 2,-3: DRAW 2,1: DRAW 4,2: DRAW 2,2
  93 PLOT 100,44: DRAW 1,6: DRAW 5,0: DRAW -1,-6: DRAW -5,0
  94 PLOT 100,44: DRAW 1,8: DRAW 7,0: DRAW -1,-8: DRAW -5,0
  95 PLOT 163,76: DRAW 0,4: DRAW 2,0: DRAW 0,-4: DRAW -2,0
  96 PLOT 144,75: DRAW 0,3: DRAW 14,1: DRAW 0,-3: DRAW -14,-1
  97 PLOT 146,78: DRAW 0,7: DRAW 2,3: DRAW 2,1: DRAW 4,1: DRAW 3,-2: DRAW 0,-9
  98 PLOT 148,78: DRAW 0,7: DRAW 2,3: DRAW 2,1: DRAW 4,1
  99 PLOT 140,80: DRAW 1,18: DRAW 26,4: DRAW 17,-4
 100 DRAW -17,4: DRAW -1,-4: DRAW 0,-18
 101 DRAW 3,-12: DRAW 5,-4: DRAW 4,-1
 102 DRAW 0,-7: DRAW 2,1: DRAW 0,6: DRAW -2,1: DRAW 0,-8
 103 DRAW 2,-9: DRAW 3,0: DRAW -2,9: DRAW -3,0
 104 DRAW 2,-9: DRAW 9,0: DRAW 1,-4: DRAW 2,0: DRAW 1,5: DRAW 5,0
 105 DRAW 1,-4: DRAW 2,0: DRAW 1
```

chas horas de trabajo, como puede verse en los listados. Pero no son necesarios para el cuarto y último programa, por lo que si "quieres ir al grano", puedes empezar directamente por el último.

Las pantallas 1 y 2 corresponden al primer programa (aunque en la primera aparece el mensaje "Pulse '1'", en realidad se puede presionar cualquier tecla). La pantalla 2 muestra a modo de pergamino las cuatro únicas operaciones del programa: izquierda, derecha, arriba y salto.

El programa 2 dibuja el primer pitufo a gran tamaño (¡Ideal para sacar una copia!), apareciendo después la palabra SMURF, es decir, pitufo en inglés. Después se carga el programa 3.

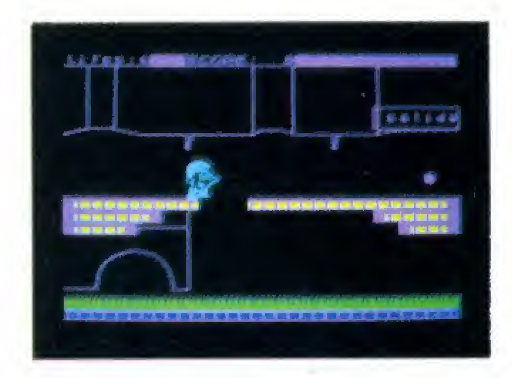
Tercer y último programa, y no precisamente corto, contiene el Pitufojuego... ¡Para pitufar toda una tarde! Las restantes pantallas muestran el laberinto que hay que recorrer para cargar la cesta con las apetitosas pitufresas, sin olvidar los pitufopeligros que acechan en cada pituforrincón. Aunque las instrucciones del programa hacen referencia a unas pitufollamas, tampoco hay que olvidar los pitufocuadros móviles, que hacen el mismo pitufodaño. Llegando al final del juego, lo cual no resulta demasiado difícil después de un poco de práctica, se puede entrar en el ranking de los mejores.

Prácticamente realizado con instrucciones PLOT y DRAW, el último programa contiene algunos caracteres gráficos definidos por el usuario, los cuales han sido subrayados para una mejor identificación.

Que haya mucha pitufosuerte y no os preocupéis: ¡pitufobromista no interviene!

```
,5: DRAW 5,0: DRAW 5,-31
 106 DRAW 12,2: DRAW -4,28: DRAW
 -12,1
 107 PLOT 132,30: DRAW 7,-1: DRA
W 0,3: DRAW 4,0: DRAW 0,-3: DRAW
 -4,0
 108 DRAW 4,0: DRAW 1,16: DRAW 2
,4: DRAW 4,1: DRAW 3,0: DRAW 4,-
2: DRAW 1,-4: DRAW 0,-17
 109 DRAW 4,0: DRAW 0,3: DRAW -4
,0
 110 PLOT 162,26: DRAW 49,-8
 111 PLOT 142,26: DRAW 0,2: DRAW
 14,-1: DRAW 0,-2: DRAW -14,1: D
RAW 14,-1: DRAW 2,1: DRAW 0,2: D
RAW -2,-1
 118 POKE 23659,0: POKE 23659,2
 120 PLOT 145,29: DRAW 1,16: DRA
W 2,4: DRAW 4,1: DRAW 3,0
 121 POKE 23659,0: POKE 23659,2
```

```
 122 PLOT 184,97: DRAW 2,-19: DR
AW 2,-12: DRAW 3,-2: DRAW 2,0: D
RAW 0,-6: DRAW 4,-10
 123 PLOT 170,76: DRAW 2,-1: DRA
W 0,4: DRAW -2,1: DRAW 0,-4
 124 PLOT 144,97: DRAW -1,4: DRA
W 3,0: DRAW -1,-4
 126 PLOT 160,96: DRAW -1,4: DRA
W 3,0: DRAW -1,-4
 127 PLOT 172,96: DRAW -1,4: DRA
W 3,0: DRAW -1,-4
 128 PLOT 137,99: DRAW 30,5: DRA
W 20,-5: DRAW -23,30: DRAW 3,-34
: DRAW -3,34: DRAW -28,-30
 130 PLOT 0,42: DRAW 5,7: DRAW 5
,2: DRAW 9,1: DRAW 7,-1: DRAW 20
,-3
 131 PLOT 46,44: DRAW 2,7: DRAW
4,1: DRAW 7,2: DRAW 33,-7
 132 PLOT 0,29: DRAW 90,10
 134 PLOT 86,0: DRAW 54,29
 135 PLOT 136,0: DRAW 26,26
 136 POKE 23659,0: POKE 23659,2
 140 PLOT 224,30: DRAW 30,-5
 141 PLOT 216,48: DRAW 4,3: DRAW
 4,1:: DRAW 2,4: DRAW 4,2: DRAW
12,2: DRAW 12,6
 142 PRINT AT 21,0;"Pulse ""1"""
 143 PAUSE 4000
 144 IF INKEY$<>"" THEN  CLS
 148 POKE 23659,0: POKE 23659,2
 150 PRINT AT 3,10;"INSTRUCCIONE
S";AT 5,8;"IZQUIERDA....Q";AT 7,
8;"DERECHA......W";AT 9,8;"ARRIB
A.......E";AT 13,8;"SALTO....T"
 151 PLOT 80,140: DRAW 104,0
 152 POKE 23659,0: POKE 23659,2
 153 PLOT 40,50: DRAW 2,-4: DRAW
 6,-5: DRAW 7,-1: DRAW 120,0
 154 DRAW 5,3: DRAW 4,7: DRAW 9,
50: DRAW 8,49: DRAW 2,4: DRAW 5,
1
 155 DRAW 3,-5: DRAW 1,-7: DRAW
-12,0
 156 PLOT 40,50: DRAW 120,0: DRA
W 2,-4: DRAW 6,-5: DRAW 7,-1
 157 PLOT 206,155: DRAW -130,0:
DRAW -6,-1: DRAW -4,-9: DRAW -22
,-95
 158 PRINT AT 21,2;"Pulse cúalqu
ier tecla"
 159 PAUSE 0: IF INKEY$<>"" THEN
  CLS
 160 PAPER 5: CLS : INK 1: PRINT
 AT 0,9;"SMURF SIDE 1": RETURN
 166 POKE 23659,0: POKE 23659,2
 170 PRINT ""
 172 POKE 23659,0: POKE 23659,2
 180 GO SUB 160
 190 POKE 23659,0: POKE 23659,2
 200 CLS : PAPER 0:: INK 3: PLOT
 0,175: DRAW 255,0: DRAW 0,-70:
DRAW -255,0: DRAW 0,70
 201 PLOT 0,76: DRAW 255,0: DRAW
 0,-70: DRAW -255,0: DRAW 0,70
 202 PLOT 0,101: DRAW 255,0: DRA
W 0,-21: DRAW -56,0: DRAW 0,17:
DRAW -149,0: DRAW 0,-17: DRAW -5
0,0: DRAW 0,21
 203 PRINT  PAPER 3; INK 7;AT 10
,7;"A.LOPEZ PRESENTA:";AT 19,9;"
PARE LA CINTA"
 204 POKE 23659,0: POKE 23659,2
 206 POKE 23659,0: POKE 23659,2
 210 FOR t=0 TO 254: OUT 254,t:
NEXT t
 220 POKE 23659,0: POKE 23659,2
 230 PAUSE 20: RETURN
```

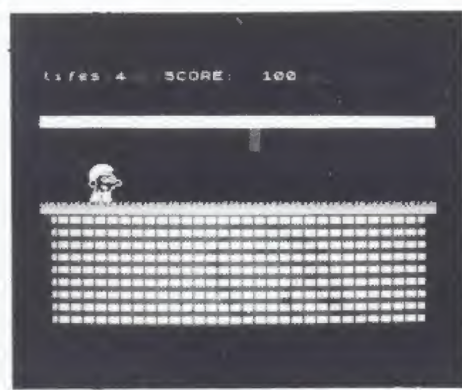


```
 231 POKE 23659,0: POKE 23659,2
 240 SAVE "SMURF" LINE 260: STOP

 250 POKE 23659,0: POKE 23659,2
 260 GO SUB 160: LOAD "logo"SCRE
EN$
 261 IF INKEY$<>"1" OR INKEY$="1
" THEN  STOP
 265 POKE 23659,0: POKE 23659,2
 270 PRINT AT 1.8:"PARE LA CINTA
.": GO TO 1
```

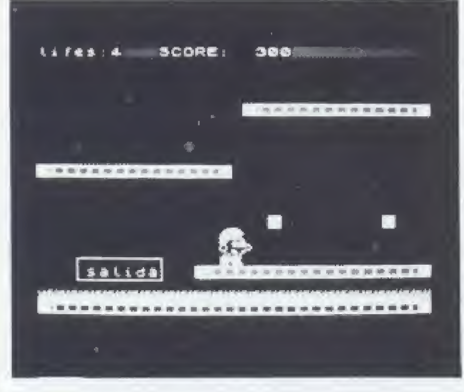
PROGRAMA 2

```
  1 PAPER 0
  2 BORDER 0
  5 CLS
  6 BORDER 0: PAPER 0
  8 INK 5
  9 PRINT  PAPER 5; INK 0;AT 0,
2;"NO PARE LA"; FLASH 1;" CINTA"
 10 PLOT 120,80: DRAW 1,2: DRAW
 2,1: DRAW 2,0: DRAW 3,-1
 11 DRAW 2,-1: DRAW 1,0: DRAW 2
,1: DRAW 19,0
 12 PLOT 120,80: DRAW 1,-2: DRA
W 2,-1: DRAW 2,0: DRAW -1,1
 13 DRAW 1,-2
 14 DRAW -1,-1
 15 DRAW -1,-2
 16 DRAW 2,-4
 17 DRAW 2,-1: DRAW 1,0: DRAW 1
,2
 18 DRAW -1,3
 19 DRAW 5,1: DRAW 2,1: DRAW 1,
3
 20 DRAW -4,-4: DRAW 18,-2
 21 DRAW 2,4: PLOT 149,71: DRAW
-3,-15
 22 DRAW 0,-2: DRAW 1,-4: DRAW
3,-1
 23 DRAW 1,0: DRAW -3,-5
 24 DRAW -1,-6
 25 DRAW 5,-4
 26 DRAW -6,0
 27 PLOT 146,20: DRAW 0,14,-PI
 28 PLOT 146,20: DRAW 12,0: DRA
W 4,2: DRAW 2,6: DRAW 0,1: DRAW
-3,5
 29 PLOT 158,75: DRAW -2,-5
 30 DRAW -3,-18
 31 DRAW 0,-2: DRAW 1,-5
 32 DRAW 4,-2: DRAW 4,0: DRAW 3
,2
 33 DRAW 1,2: DRAW -1,1: DRAW -
2,-1: DRAW -1,1
 34 PLOT 165,48: DRAW 1,2: DRAW
 -1,1: DRAW -2,-1: DRAW -1,1
 35 PLOT 165,52: DRAW 1,2: DRAW
 0,1: DRAW -1,1: DRAW -2,-1: DRA
W -2,0
 36 DRAW 1,0: DRAW 3,22
 37 DRAW 0,-20
 38 PLOT 160,35: DRAW -3,1: DRA
W -2,3: DRAW 0,1: DRAW 4,1: DRAW
 3,2
 39 PLOT 147,58: DRAW 6,0
 40 PLOT 138,49: DRAW 8,2
 41 PLOT 138,49: DRAW -1,5: DRA
W -12,0,PI: DRAW 0,-3: DRAW 1,-4
 42 DRAW 1,-1: DRAW 2,-9: DRAW
3,-1: DRAW 3,0: DRAW 4,3
 43 DRAW 1,2
 50 PLOT 140,42: DRAW 6,2
 51 PLOT 150,85: DRAW 4,-2: DRA
W 2,-1: DRAW 6,-1: DRAW 3,1
 52 DRAW 4,1
 53 DRAW 3,-1: DRAW 3,1: DRAW 1
,3: DRAW 0,2: DRAW -1,2: DRAW -4
,1
 54 DRAW -2,-1
 55 PLOT 170,86: DRAW 2,1: DRAW
 2,-1: DRAW -1,1: DRAW -2,-1
 56 PLOT 170,91: DRAW -1,6
 57 PLOT 175,91: DRAW 2,6: DRAW
 0,2
 58 DRAW -1,8: DRAW -2,4: DRAW
-6,8
 59 DRAW -4,2
 60 DRAW -6,1: DRAW -6,-1: DRAW
-5,-3
 61 DRAW -1,-3
 62 DRAW 0,-3: DRAW 4,-1: DRAW
5,-1: DRAW -2,1: DRAW -2,-3
 63 DRAW 9,-1: DRAW -9,1: DRAW
-3,-1
 66 PLOT 146,87: DRAW 3,-1
 67 PLOT 145,88: DRAW 6,-1
 68 DRAW -6,1: DRAW 0,9,-PI
 69 DRAW 2,0: DRAW 4,-3
 70 PLOT 154,87: DRAW 2,-1: DRA
W 5,1: DRAW 4,3: DRAW -2,1: DRAW
 4,-1
 71 PLOT 157,[illegible]: DRAW 1,-2: DRA
W 4,2: DRAW 1,2: DRAW -5,-1: DRA
W 0,-1: DRAW 4,0: DRAW -3,0: DRA
W 0,-1
 72 PLOT 152,96: DRAW 1,2: DRAW
 1,1: DRAW 1,0: DRAW 1,-1: DRAW
-2,0: DRAW 2,0: DRAW -1,-2: DRAW
 -2,-1: DRAW 1,2
```




```
 73 PLOT 147,97: DRAW 1,2: DRAW
 1,1: DRAW 1,0: DRAW 1,-1: DRAW
-2,0: DRAW 2,0: DRAW -1,-2: DRAW
 -2,-1: DRAW 1,2
 74 PLOT 156,102: DRAW 2,0: DRA
W 3,-2
 75 PLOT 145,102: DRAW 3,1: DRA
W 2,0
 80 PLOT 144,97: DRAW -1,5: DRA
W 3,3
 90 PLOT 156,121: DRAW 8,0
 91 DRAW 1,-1: DRAW -13,0
 92 DRAW -2,-1: DRAW 17,0
 93 DRAW 1,-1: DRAW -20,0
 94 DRAW -1,-1: DRAW 22,0
 95 DRAW 1,-1: DRAW -23,0
 96 DRAW -1,-1: DRAW 24,0
 97 DRAW 2,-1: DRAW -25,0
 98 DRAW 0,-1: DRAW 25,0
 99 DRAW 1,-1: DRAW -25,0
100 DRAW 0,-1: DRAW 25,0
101 DRAW 1,-1: DRAW -18,0
102 DRAW 0,-1: DRAW 18,0
103 DRAW 0,-1: DRAW -14,0
104 DRAW 2,-1: DRAW 14,0
105 DRAW 0,-1: DRAW -14,0
106 DRAW 1,-1: DRAW 13,0
107 DRAW 0,-1: DRAW -12,0
108 DRAW 1,-1: DRAW 11,0
109 DRAW 0,-1: DRAW -9,0
110 DRAW 1,-1: DRAW 9,0
111 DRAW 0,-1: DRAW -8,0
112 DRAW 0,-1: DRAW 8,0
113 DRAW 0,-1: DRAW -7,0
114 DRAW 0,-1: DRAW 7,0
115 DRAW -1,-1: DRAW -6,0
117 DRAW 0,-1: DRAW 6,0
118 DRAW 0,-1: DRAW -6,0
119 DRAW 1,-1: DRAW 5,0: DRAW 0
,-1: DRAW -2,0
120 PLOT 129,59: DRAW 4,0
121 DRAW 1,-1: DRAW -6,0
122 DRAW -1,-1: DRAW 8,0
123 DRAW 1,-1: DRAW -10,0
124 DRAW 0,-1: DRAW 10,0: DRAW
0,-1: DRAW -10,0
125 DRAW 0,-1: DRAW 10,0: DRAW
0,-1: DRAW -10,0
126 DRAW 0,-1: DRAW 11,0: DRAW
0,-1: DRAW -11,0
127 DRAW 1,-1: DRAW 11,0
128 DRAW 1,-1: DRAW -12,0
129 DRAW 0,-1: DRAW 12,0
130 DRAW 1,-1: DRAW -12,0
131 DRAW 0,-1: DRAW 12,0: DRAW
0,-1: DRAW -12,0
132 DRAW 1,-1: DRAW 12,0
133 DRAW 1,-1: DRAW -13,0: DRAW
-1,-1: DRAW 1.,0
134 DRAW 1,-1: DRAW -11,0: DRAW
-1,-1: DRAW 9,0
135 DRAW -1,-1: DRAW -8,0: DRAW
3,-1: DRAW 4,0
137 PLOT 138,49: DRAW 7,2: DRAW
1,-1: DRAW -7,-2: DRAW 0,-2: DR
AW 8,0: DRAW 0,-1: DRAW -8,0: DR
AW 0,-1: DRAW 5,0
140 PLOT 146,57: DRAW 8,0
141 DRAW -1,0: DRAW 0,-1: DRAW
-7,0
142 DRAW 0,-1: DRAW 7,0: DRAW 0
,-1: DRAW -7,0
143 DRAW 0,-1: DRAW 7,0: DRAW 0
,-1: DRAW -6,0
144 DRAW 0,-1: DRAW 6,0: DRAW 0
,-1: DRAW -4,0
145 PLOT 150,49: DRAW 3,0: DRAW
0,-1: DRAW -3,0: DRAW 1,-1: DRA
W 3,0
146 DRAW 0,-1: DRAW -4,0
147 DRAW 0,-1: DRAW 4,0
148 DRAW 1,-1: DRAW -6,0
149 DRAW -1,-1: DRAW 8,0
150 DRAW 3,0: DRAW -1,-1:: DRAW
-9,0
151 DRAW -1,-1: DRAW 6,0
152 DRAW 0,-1: DRAW -6,0
153 DRAW 0,-1: DRAW 6,0
154 DRAW 1,-1: DRAW -7,0
155 DRAW 1,-1: DRAW 7,0: DRAW 1
,-1: DRAW -7,0
156 DRAW 1,-1: DRAW 6,0
160 DRAW 2,-1: DRAW -14,0
161 DRAW 0,-1: DRAW 16,0
162 DRAW 1,-1: DRAW -20,0
163 DRAW -1,-1: DRAW 21,0
164 DRAW 1,-1: DRAW -23,0
165 DRAW -1,-1: DRAW 24,0
166 DRAW 1,-1: DRAW -25,0
167 DRAW 0,-1: DRAW 24,0
168 DRAW 0,-1: DRAW -24,0
169 DRAW 1,-1: DRAW 22,0
170 DRAW 0,-1: DRAW -22,0
171 DRAW 1,-1: DRAW 21,0
178 DRAW 0,-1: DRAW -20,0
```

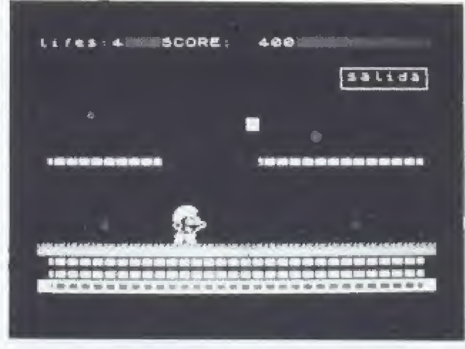
```
179 DRAW 0,-1: DRAW 16,0
180 PLOT  INVERSE 1;153,110: DRAW  INVERSE 1;7,1
190 INVERSE 0: INK 6
192 PLOT 20,40: DRAW 1,-4: DRAW 2,-3: DRAW 6,-2: DRAW 49,0
193 DRAW 4,3: DRAW 1,4: DRAW 9,50
194 DRAW 4,2: DRAW 3,0: DRAW 4,-2: DRAW 1,-5: DRAW -12,0
195 PLOT 20,40: DRAW 47,0: DRAW 1,-4: DRAW 2,-3: DRAW 3,-2
196 PLOT 25,40: DRAW 4,3: DRAW 1,4: DRAW 9,40: DRAW 4,3: DRAW 4,1: DRAW 48,0
200 PLOT 48,80: DRAW 0,1: DRAW 1,2: DRAW -3,0: DRAW 6,0
201 PLOT 55,80: DRAW 0,1: DRAW 1,2: DRAW -1,-1: DRAW 3,0: DRAW 1,1: DRAW -1 -2: DRAW 0,-1
202 PLOT 65,80: DRAW -4,0: DRAW 0,2: DRAW 2,0: DRAW -2,0: DRAW 1,2: DRAW 3,0
203 PLOT 38,67: DRAW 0,1: DRAW
```



```
1,2: DRAW 2,-2: DRAW 2,2: DRAW 1,-2: DRAW 0,-1
204 PLOT 46,67: DRAW 0,1: DRAW 1,1: PLOT 48,71
205 PLOT 48,67: DRAW 2,0: DRAW 1,2: DRAW -2,0: DRAW 1,2: DRAW 2,0
206 PLOT 56,67: DRAW 0,1: DRAW 1,2: DRAW -3,0: DRAW 6,0
207 PLOT 64,67: DRAW -4,0: DRAW 0,2: DRAW 2,0: DRAW -2,0: DRAW 1,2: DRAW 3,0
208 PLOT 66,67: DRAW 0,1: DRAW 1,2: DRAW 4,0: DRAW 0,-1: DRAW -1,-1: DRAW -1,0: DRAW 1,-2
209 PLOT 72,67: DRAW 0,1: DRAW 1,1: PLOT 74,71
210 PLOT 74,66: DRAW 2,0: DRAW 1,4: DRAW -2,0: DRAW -1,-4
211 PLOT 79,70: DRAW -1,-4: DRAW 2,0: DRAW 1,4
212 PLOT 82,66: DRAW 2,0: DRAW 1,2: DRAW -2,0: DRAW 1,2: DRAW 2,0
213 PLOT 40,50: DRAW -3,0: DRAW 0,1: DRAW 1,3: DRAW 2,0
214 PLOT 43,50: DRAW 0,1: DRAW 1,3: DRAW 2,0: DRAW 1,-3: DRAW 0,-1: DRAW 0,1: DRAW -4,0
215 PLOT 50,50: DRAW 2,0: DRAW 1,2: DRAW -2,0: DRAW 1,2: DRAW 2,0
216 PLOT 58,50: DRAW 0,1: DRAW 1,3: DRAW -3,0: DRAW 6,0
217 PLOT 67,50: DRAW -4,0: DRAW 0,1: DRAW 1,3
220 PLOT 73,50: DRAW -4,0: DRAW 0,2: DRAW 2,0: DRAW -2,0: DRAW 1,2: DRAW 3,0
230 INK 5
240 PLOT 0,8: DRAW 142,0: DRAW 0,10: DRAW 25,0: DRAW 0,10: DRAW 88,0
241 PLOT 0,4: DRAW 145,0: DRAW 0,10: DRAW 25,0: DRAW 0,10: DRAW 85,0
250 PLOT 190,32: DRAW 0,12: DRAW 17,0: DRAW 0,-12: DRAW -17,0
251 PLOT 190,52: DRAW 0,12: DRAW 17,0: DRAW 0,-12: DRAW -17,0
252 PLOT 190,72: DRAW 0,12: DRAW 17,0: DRAW 0,-12: DRAW -17,0
253 PLOT 190,92: DRAW 0,12: DRAW 17,0: DRAW 0,-12: DRAW -17,0
254 PLOT 190,112: DRAW 0,12: DRAW 17,0: DRAW 0,-12: DRAW -17,0
255 PLOT 190,132: DRAW 0,12: DRAW 17,-3: DRAW 0,-9: DRAW -17,0
256 PLOT 195,150: DRAW 4,12: DRAW 17,-7: DRAW -5,-9: DRAW -17,4
257 PLOT 202,164: DRAW 15,-5: DRAW 6,0: DRAW 0,9
258 DRAW -20,-3
259 PLOT 230,159: DRAW 0,9: DRAW 17,0: DRAW 0,-9: DRAW -17,0
260 PLOT 251,159: DRAW 0,9: DRAW 4,0: DRAW -4,0: DRAW 0,-9: DRAW 4,0
270 PLOT 211,29: DRAW 0,100
271 DRAW 1,7: DRAW 2,6
272 DRAW 8,9: DRAW 4,1
273 DRAW 29,0
280 PLOT 70,96: DRAW 0,12: DRAW 45,0: DRAW 0,-12: DRAW -45,0
281 PLOT 74,109: DRAW 0,30: DRAW 1,4: DRAW 2,4: DRAW 4,2: DRAW 4,2: DRAW 15,0: DRAW 4,-2: DRAW 4,-2: DRAW 2,-4: DRAW 1,-4: DRAW 0,-30
282 PLOT 77,109: DRAW 0,26: DRAW 1,4: DRAW 2,4: DRAW 4,2: DRAW 4,2: DRAW 15,0: DRAW 4,-2: DRAW 4,-2
288 FOR y=0 TO 1: FOR t=0 TO 68: OUT 254,t: BEEP .0030*INT (RND*12),t+RND: NEXT t: NEXT y
300 CLS
305 PRINT  INK 3;AT 9,7;"[illegible]"
306 PRINT  INK 6;AT 10,7;"[illegible]"
307 PRINT  INK 3;AT 11,7;"[illegible]"
310 LOAD "3"
```

LISTADO 3

```
   1 CLS : PAPER 0: BORDER 0
   2 PAPER 0: BORDER 0
   3 POKE USR "s"+0,BIN 11111111: POKE USR "s"+1,BIN 11111111: POKE USR "s"+2,BIN 00110000: POKE USR "s"+3,BIN 00110000: POKE USR "s"+4,BIN 00110000: POKE USR "s"+5,BIN 00110000: POKE USR "s"+6,BIN 11111111: POKE USR "s"+7,BIN 11111111
   4 BORDER 0: PAPER 0: POKE USR "u"+0,BIN 01001001: POKE USR "u"+1,BIN 10011011: POKE USR "u"+2,BIN 11011011: POKE USR "u"+3,BIN 01101101: POKE USR "u"+4,BIN 11111111: POKE USR "u"+5,BIN 11111111: POKE USR "u"+6,BIN 11111111: POKE USR "u"+7,BIN 11111111
   5 POKE USR "t"+0,BIN 00011000: POKE USR "t"+1,BIN 01111110: POKE USR "t"+2,BIN 01111110: POKE USR "t"+3,BIN 11111111: POKE USR "t"+4,BIN 11111111: POKE USR "t"+5,BIN 01111110: POKE USR "t"+6,BIN 01111110: POKE USR "t"+7,BIN 00011000
   6 FOR t=USR "a" TO USR "j": READ x: POKE t,x: NEXT t
   8 POKE USR "j"+0,BIN 00000001: POKE USR "j"+1,BIN 00000001: POKE USR "j"+2,BIN 00001110: POKE USR "j"+3,BIN 00010001: POKE USR "j"+4,BIN 00010000: POKE USR "j"+5,BIN 00001110: POKE USR "j"+6,BIN 00000001: POKE USR "j"+7,BIN 00000000
  10 DATA 102,153,186,119,0,128,225,243
  11 DATA 7,31,63,127,127,255,255,252: DATA 240,252,254,255,206,240,254,1: DATA 248,240,112,136,136,136,112,8
  12 DATA 128,128,112,136,8,112,128,0
  13 DATA 102,153,93,238,0,1,7,1
```



```
5: DATA 31,15,14,17,17,17,14,16: DATA 224,248,252,254,254,255,255,63: DATA 15,63,127,255,115,15,127,128: DATA 1,1,14,17,16,14,1,0
  14 POKE USR "o"+0,BIN 00010010
  15 POKE USR "o"+1,BIN 00010010
  16 POKE USR "o"+2,BIN 00111111
  17 POKE USR "o"+3,BIN 01011111
  18 POKE USR "o"+4,BIN 00111110
  20 POKE USR "o"+5,BIN 01111110
  21 POKE USR "o"+6,BIN 11111110
  22 POKE USR "o"+7,BIN 01111111
  30 POKE USR "p"+0,BIN 00100100
  31 POKE USR "p"+1,BIN 00100100
  32 POKE USR "p"+2,BIN 11111100
  33 POKE USR "p"+3,BIN 11111100
  34 POKE USR "p"+4,BIN 01111000
  35 POKE USR "p"+5,BIN 11111100
  37 POKE USR "p"+6,BIN 11111110
  38 POKE USR "p"+7,BIN 01111100
  40 POKE USR "q"+0,BIN 00010010
  41 POKE USR "q"+1,BIN 00100100
  42 POKE USR "q"+2,BIN 01111111
  43 POKE USR "q"+3,BIN 10111111
  44 POKE USR "q"+4,BIN 01111110
  45 POKE USR "q"+5,BIN 01111100
  46 POKE USR "q"+6,BIN 11111000
  47 POKE USR "q"+7,BIN 01111100
  50 POKE USR "r"+0,BIN 00100100
  51 POKE USR "r"+1,BIN 00010010
  52 POKE USR "r"+2,BIN 11111100
  53 POKE USR "r"+3,BIN 01111100
  54 POKE USR "r"+4,BIN 00111100
  55 POKE USR "r"+5,BIN 00111110
  56 POKE USR "r"+6,BIN 00011111
  57 POKE USR "r"+7,BIN 00111111
  60 POKE USR "k"+0,BIN 01001000: POKE USR "k"+1,BIN 01001000: POKE USR "k"+2,BIN 11111100: POKE USR "k"+3,BIN 11111010: POKE USR "k"+4,BIN 01111100: POKE USR "k"+5,BIN 01111110: POKE USR "k"+6,BIN 01111111: POKE USR "k"+7,BIN 11111110
  62 POKE USR "l"+0,BIN 00100100: POKE USR "l"+1,BIN 00100100: P
```

Programas

43

```
OKE USR "l"+2,BIN 00111111: POKE
 USR "l"+3,BIN 00111111: POKE US
R "l"+4,BIN 00011110: POKE USR "
l"+5,BIN 00111111: POKE USR "l"+
6,BIN 01111111: POKE USR "l"+7,B
IN 00111110
  70 POKE USR "m"+0,BIN 00100100
: POKE USR "m"+1,BIN 00010010: P
OKE USR "m"+2,BIN 00111111: POKE
 USR "m"+3,BIN 00111110: POKE US
R "m"+4,BIN 00111100: POKE USR "
m"+5,BIN 01111100: POKE USR "m"+
6,BIN 11111000: POKE USR "m"+7,B
IN 11111100
  80 POKE USR "n"+0,BIN 00100100
: POKE USR "n"+1,BIN 01001000: P
OKE USR "n"+2,BIN 00111111: POKE
 USR "n"+3,BIN 00111110: POKE US
R "n"+4,BIN 00111100: POKE USR "
n"+5,BIN 01111100: POKE USR "n"+
6,BIN 11111000: POKE USR "n"+7,B
IN 11111100
 300 INK 5: PRINT AT 5,0; INK 5;
"EL MALVADO GARGAMELHA ROBADO LA
S PITUFRESAS("; INK 3;"I"; INK 5
;") DE LOS PITUFOS."
 301 PRINT INK 5;AT 9,1;"DEBES R
ECOGER LAS QUE PUEDAS,   TENIEND
O CUIDADO CON LAS LLAMAS("; INK
3;"U"; INK 5;")."
 320 PRINT INK 6;AT 21,6;"PULSE
CUALQUIER TECLA": PAUSE 0
 340 CLS
 344 INK 5: PRINT FLASH 1; INK 6
;AT 1,0;"Y RECUERDA..."
 346 INK 5: PLOT 25,70: DRAW 0,7
5: DRAW 116,0: DRAW 0,-75: DRAW
-116,0
```




```
 352 PRINT INK 6;AT 21,6;"PULSE
CUALQUIER TECLA": PAUSE 0
 360 PAUSE 0
 390 CLS
 392 PAPER 0: BORDER 0
 400 LET a$=" BC "
 401 LET b$=" DBE "
 402 LET c$=" OP "
 403 LET d$=" QR "
 404 LET f$="  IH "
 405 LET g$=" JFG "
 406 LET h$="  LK "
 407 LET i$="  NR "
 414 LET t=9: LET r=3
 420 PRINT PAPER 6; INK 3;AT 12,
0;"█SSSSSSSSSSSSSSSSSSSSSSSSSSSSS
SS█"
 421 PRINT PAPER 6; INK 3;AT 13,
0;"█SSSSSS█";AT 13,25;"█SSSSS█"
 422 PRINT PAPER 6; INK 3;AT 14,
0;"█SSSS█";AT 14,27;"█SSS█"
 430 PRINT INK 4;AT 20,0;"UUUUUU
UUUUUUUUUUUUUUUUUUUUUUUUUU"; PAP
ER 1; INK 3;;AT 21,0;"SSSSSSSSSS
SSSSSSSSSSSSSSSSSSSSSS"
 440 INK 3: PLOT 0,120: DRAW 12,
3: DRAW 6,1: DRAW 12,0: DRAW 19,
-4: DRAW 30,0: DRAW 0,-9: DRAW 2
,0: DRAW 1,9: DRAW 40,0: DRAW 7,
3: DRAW 12,0: DRAW 9,-3
 441 DRAW 24,0: DRAW 0,-9: DRAW
2,0: DRAW 1,9: DRAW 24,0
 442 DRAW 15,3: DRAW 39,0
 443 PLOT 15,125: DRAW 0,40: PLO
T 33,125: DRAW 0,42
 444 PLOT 0,20: DRAW 20,0: DRAW
50,0,-PI: DRAW 10,0: DRAW 0,50
 445 PLOT 123,120: DRAW 0,50
 446 PLOT 149,120: DRAW 0,50
 447 PLOT 203,120: DRAW 0,50
```

```
 450 PLOT 40,60: DRAW 40,0
 351 PRINT INK 6;AT 5,4;"IZQUIER
DA...Q";AT 7,4;"DERECHA....W";AT
 9,4;"ARRIBA.....E";AT 11,4;"SAL
TO.......T"
 460 PRINT INK 3;AT 0,0;"████████
████████████████████████"
 480 LET f=4: PRINT AT 0,0;"life
s: ";f
 482 LET c=0: PRINT AT 0,10;"SCO
RE:  ";c
 483 PLOT INK 3;0,167: DRAW INK
3;255,0
 485 PRINT AT 5,26;"salida": PLO
T 200,128: DRAW 0,10: DRAW 55,0:
 DRAW 0,-14: DRAW -55,0: DRAW 0,
4
 490 PRINT INK 3;AT 10,29;"I"
 500 INK 5: IF INKEY$="" THEN PR
INT AT t,r;a$;AT t+1,r;b$;AT t+2
,r;c$
 510 IF INKEY$="w" OR INKEY$="W"
 THEN LET r=r+1: PRINT AT t,r;a$
;AT t+1,r;b$;AT t+2,r;c$: PRINT
AT t+2,r;d$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;c$
 520 IF INKEY$="q" OR INKEY$="Q"
 THEN LET r=r-1: PRINT AT t,r;f$
;AT t+1,r;g$;AT t+2,r;h$: PRINT
AT t+2,r;i$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;h$
 522 IF r=0 THEN LET r=r+1: PRIN
T AT t+1,r;" "
 523 IF r=28 THEN LET r=r-1: PRI
NT AT t+1,r;" "
 526 IF r=8 THEN PRINT INK 0;AT
12,11;"    "
 528 POKE 23658,0: IF r=9 AND IN
KEY$="t" THEN PRINT AT 9,10;"  "
;AT 10,10;"    ";AT 11,10;"   ": P
RINT AT t,r+2;a$;AT t+1,r+2;b$;A
T t+2,r+2;c$: PRINT AT 9,12;"  "
;AT 10,12;"    ";AT 11,12;"   ": L
ET t=9: LET r=14
 530 IF r=10 AND NOT INKEY$="t"
THEN PRINT AT 9,11;"    ";AT 10,1
1;"    ";AT 11,11;"    ": LET t=
17: LET r=11: PRINT AT t,r;a$;AT
 t+1,r;b$;AT t+2,r;c$
 532 IF t=17 AND r=11 THEN FOR g
=16 TO 19: FOR y=0 TO 2: PRINT A
T g,12;"████";AT g-1,12;"    ":
 NEXT g: FOR g=0 TO 40: BEEP .03
5,g+RND: NEXT g: LET f=f-1: PRIN
T AT 0,6;f: PRINT AT 19,12;"
 ": PRINT PAPER 6; INK 3;AT 12,1
1;"SSSS": LET t=9: LET r=3
 534 IF r=11 AND NOT INKEY$="t"
THEN PRINT AT 9,13;"    ";AT 10,1
3;"    ";AT 11,13;"    ": LET t=
17: LET r=11: PRINT AT t,r;a$;AT
 t+1,r;b$;AT t+2,r;c$
 535 IF r=12 AND NOT INKEY$="t"
THEN PRINT AT 9,13;"    ";AT 10,1
3;"    ";AT 11,13;"    ": LET t=
17: LET r=11: PRINT AT t,r;a$;AT
 t+1,r;b$;AT t+2,r;c$
 537 IF r=18 AND NOT INKEY$="t"
THEN PRINT AT 9,19;"    ";AT 10,1
9;"    ";AT 11,19;"    ": LET t=
17: LET r=19: PRINT AT t,r;a$;AT
 t+1,r;b$;AT t+2,r;c$
 540 IF r=16 THEN PRINT AT 12,19
;"    "
 542 IF r=17 AND INKEY$="t" THEN
 PRINT AT 9,18;"  ";AT 10,18;"
";AT 11,18;"   ": PRINT AT t,r+2;
a$;AT t+1,r+2;b$;AT t+2,r+2;c$:
PRINT AT 9,20;"  ";AT 10,20;"
";AT 11,20;"   ": LET t=9: LET r=
22
 544 IF r=20 AND NOT INKEY$="T"
THEN PRINT AT 9,20;"    ";AT 10,
20;"    ";AT 11,20;"    ": LET t
=17: LET r=19: PRINT AT t,r;a$;A
T t+1,r;b$;AT t+2,r;c$
 546 IF t=17 AND r=19 THEN FOR g
=16 TO 19: FOR y=0 TO 2: PRINT A
T g,19;"████";AT g-1,19;"    ":
 NEXT g: FOR g=0 TO 40: BEEP .03
5,g+RND: NEXT g: LET f=f-1: PRIN
T AT 0,6;f: PRINT ;AT 19,19;"
";AT 19,12;"    ": PRINT PAPER
 6; INK 3;AT 12,11;"SSSS";AT 12,
19;"SSSS": LET t=9: LET r=3
 550 IF r=25 AND INKEY$="w" THEN
 LET c=c+100: PRINT INK 6;AT 0,1
8;c: FOR g=0 TO 254 STEP 5: OUT
254,g: NEXT g: CLS : GO TO 600
 551 IF f=0 THEN CLS : GO TO 300
```

```
 560 GO TO 500
 600 CLS
 601 LET a$=" BC "
 602 LET b$=" DBE "
 603 LET c$=" OP "
 604 LET d$=" QR "
 605 LET e$="  IH "
 606 LET f$=" JFG "
 607 LET g$="  LK "
 608 LET h$="  MR "
 610 REM
 620 FOR d=12 TO 20: PRINT INK 3
; PAPER 6;AT d,0;"█SSSSSSSSSSSSS
SSSSSSSSSSSSSSSSS█": NEXT d
 621 PRINT INK 4;AT 11,0;"UUUUUU
UUUUUUUUUUUUUUUUUUUUUUUUUU"
 624 PRINT INK 3;AT 0,0;"████████
████████████████████████"
 626 PRINT AT 0,0;"lifes:";f
 628 PRINT AT 0,10;"SCORE:  ";c
 630 PLOT INK 3;0,167: DRAW INK
3;255,0
 640 INK 3: PLOT 0,146: DRAW 20,
0: DRAW 30,0,-PI: DRAW 20,0: DRA
W 30,0,-PI: DRAW 30,0: DRAW 30,0
,-PI: DRAW 30,0: DRAW 30,0,-PI:
DRAW 34,0
 641 INK 3: PLOT 0,148: DRAW 20,
0: DRAW 30,0,-PI: DRAW 20,0: DRA
W 30,0,-PI: DRAW 30,0: DRAW 30,0
,-PI: DRAW 30,0: DRAW 30,0,-PI:
DRAW 34,0
 643 PRINT AT 9,22;"I"
 647 PRINT AT 8,26;"salida": PLO
T 200,104: DRAW 0,10: DRAW 55,0:
 DRAW 0,-14: DRAW -55,0: DRAW 0,
4
 650 LET t=8: LET r=3
```




```
 655 LET x=10: LET y=26
 656 LET n=10: LET o=21
 657 LET v=5: LET z=29
 658 PRINT INK 6;AT 4,0;"████████
████████████████████████"
 660 INK 5: PRINT INK 3;AT v,z;"
█";AT v,z+1;" ";AT v+1,z;"█";AT
v+1,z+1;" ": LET z=z-1: INK 2: P
RINT AT x,y;"U";AT x,y+1;" ";AT
n,o;"U";AT n,o+1;" ": LET o=o-1:
 IF INKEY$="" THEN PRINT INK 5;A
T t,r;a$;AT t+1,r;b$;AT t+2,r;c$
 661 IF x=10 AND y=1 THEN PRINT
AT x,y;" ": LET y=y+1: LET y=28
 663 IF z=2 THEN LET z=z+1: PRIN
T INK 0;AT 5,1;"   ";AT 6,2;"
 ": LET z=30
 666 IF o=1 THEN LET o=o+1: PRIN
T AT 10,2;" ": LET o=21
 668 INK 5: IF INKEY$="e" OR INK
EY$="E" THEN LET t=6: PRINT ;AT
t+4,r+1;"   ";AT t+3,r+1;"   ";AT
 t,r;a$;AT t+1,r;b$;AT t+2,r;c$:
 FOR b=0 TO 1: NEXT b: PRINT AT
t,r;"    ";AT t+1,r;"    ";AT t+2
,r;"    ": LET t=8
 670 IF INKEY$="w" OR INKEY$="W"
 THEN LET r=r+1: PRINT AT t,r;a$
;AT t+1,r;b$;AT t+2,r;c$: PRINT
AT t+2,r;d$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;c$
 672 IF INKEY$="q" OR INKEY$="Q"
 THEN LET r=r-1: PRINT AT t,r;e$
;AT t+1,r;f$;AT t+2,r;h$: PRINT
AT t+2,r;g$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;h$
 690 IF r=0 THEN LET r=r+1: PRIN
T AT t+1,r;" "
 692 IF r=28 THEN LET r=r-1: PRI
NT AT t+1,r;" "
```

OFERTA DE SUSCRIPCION

Todospectrum

REVISTA EXCLUSIVA PARA USUARIOS

Te ofrece la posibilidad de suscribirte con unas condiciones muy ventajosas para ti:

1 *Recibir puntualmente, en tu domicilio la publicación TODOSPECTRUM que durante 12 meses te proporcionará lectura, programas, ayuda, entretenimiento, etc.*

2 *Consigue un práctico regalo:*



Una obra fundamental en la biblioteca de los aficionados al Spectrum:

- *Reglas y herramientas del Basic.*
- *La técnica de los organigramas.*
- *El fabuloso mundo de las rutinas.*
- *Variables y cadenas.*
- *Funciones matemáticas usuales.*

Esta publicación, escrita con estilo ameno y práctico, te ayudará a sacar todo el partido posible a tu máquina.

3 *La opción de ser protagonista. Tú puedes tener una participación directa con tus comentarios, programas, sugerencias, etc.*

4 *Obtener premios importantes con tus programas, y temas de interés.*

EN DEFINITIVA, TODO SON VENTAJAS

No dejes pasar esta oportunidad, suscríbete a "TODOSPECTRUM", cumplimentando hoy mismo el cupón de respuesta adjunto.



Programas

43

```
 700 LET y=y-1
 780 IF r=y AND NOT INKEY$="e" T
HEN PRINT INK 3; FLASH 1;AT t,r;
"   ";AT t+1,r;"   ";AT t+2,r;
"   ": FOR j=0 TO 40: BEEP .035
,j*RND: NEXT j: FLASH 0: PRINT F
LASH 0;AT t,r;"   "; FLASH 0;A
T t+1,r;"   "; FLASH 0;AT t+2,
r;"   ": LET t=8: LET r=3: LET
 f=f-1: PRINT AT 0,6;f: PRINT IN
K 3;AT 9,22;"I"
 781 IF r=o AND NOT INKEY$="e" T
HEN PRINT INK 3; FLASH 1;AT t,r;
"   ";AT t+1,r;"   ";AT t+2,r;
"   ": FOR j=0 TO 40: BEEP .035
,j*RND: NEXT j: FLASH 0: PRINT F
LASH 0;AT t,r;"   "; FLASH 0;A
T t+1,r;"   "; FLASH 0;AT t+2,
r;"   ": LET t=8: LET r=3: LET
 f=f-1: PRINT AT 0,6;f: PRINT IN
K 3;AT 9,22;"I"
 782 IF r=z AND INKEY$="e" THEN
PRINT INK 3; FLASH 1;AT t,r;"
 ";AT t+1,r;"   ";AT t+2,r;"
 ": FOR j=0 TO 40: BEEP .035,j*R
ND: NEXT j: FLASH 0: PRINT FLASH
 0;AT t,r;"   "; FLASH 0;AT t+
1,r;"   "; FLASH 0;AT t+2,r;"
   ": LET t=8: LET r=3: LET f=f
-1: PRINT AT 0,6;f: PRINT INK 3;
AT 9,22;"I"
 790 IF r=20 AND INKEY$="w" THEN
 LET c=c+100: PRINT AT 0,18;c: F
OR k=0 TO 254 STEP 2: OUT 254,k:
 NEXT k: CLS : GO TO 1000
 791 IF r=21 THEN CLS : GO TO 10
00
 800 IF f=0 THEN CLS : GO TO 300
 900 GO TO 660
1000 REM 3 parte
1010 LET t=2: LET r=22
1100 LET a$=" BC ": LET b$=" DAE
 ": LET c$=" QP ": LET d$=" QR "
: LET e$="  IH ": LET f$=" JFG "
: LET g$="  LK ": LET h$="  MR "
1105 PRINT INK 3;AT 0,0;"[redacted]"
1110 PRINT AT 0,0;"lifes: ";f
1120 PRINT AT 0,10;"SCORE:   ";c
1200 PRINT INK 6; PAPER 3;AT 5,1
7;"█SSSSSSSSSSSSS█"
1201 PRINT INK 6; PAPER 3;AT 10,
0;"█SSSSSSSSSSSSSS█"
1202 PRINT INK 6; PAPER 3;AT 18,
13;"█SSSSSSSSSSSSSSSSS█"
1210 PRINT PAPER 1; INK 6;AT 21,
0;"█SSSSSSSSSSSSSSSSSSSSSSSSSSSSS
SS█"
1211 PRINT INK 4;AT 20,0;"UUUUUU
UUUUUUUUUUUUUUUUUUUUUUUUUU"
1220 PRINT INK 3;AT 4,7;"I"
1221 PRINT INK 3;AT 8,3;"I"
1222 PRINT INK 3;AT 8,12;"I"
1230 PRINT INK 3;AT 3,19;"I"
1231 PRINT INK 3;AT 16,14;"I"
1232 PRINT INK 3;AT 16,27;"I"
1240 PRINT AT 18,4;"salida": PLO
T 26,26: DRAW 0,10: DRAW 55,0: D
RAW 0,-14: DRAW -55,0: DRAW 0,4
1300 LET x=9: LET y=15: PRINT IN
K 3;AT x,y;"U"
1400 LET d=7: LET g=19
1450 INK 5: PRINT INK 6;AT d,g;"
█";AT d-1,g;" ";AT d,g+9;"█";AT
d-1,g+9;" ": LET d=d+1: PRINT IN
K 2;AT x,y;"U";AT x,y+1;" ": LET
 y=y-1: IF INKEY$="" THEN PRINT
AT t,r;a$;AT t+1,r;b$;AT t+2,r;c
$
1501 IF t=15 AND r=17 AND r=d TH
EN BEEP .01,0: PRINT FLASH 0;AT
t,r+1;"   ";AT t+1,r+1;"   ";AT t
+2,r+1;"   ": PRINT INK 3;AT 2,19
;"I";AT 4,7;"I";AT 8,3;"I";AT 8,
12;"I": LET f=f-1: PRINT AT 0,6;
f: LET t=2: LET r=22
1505 IF d=18 THEN LET d=d-1: LET
 d=7: PRINT AT 17,19;"         ";AT
17,28;"  "
1506 IF y=0 THEN LET y=y+1: LET
y=15: PRINT AT 9,1;" "
1508 IF r=27 THEN LET r=r-1: PRI
NT AT t+1,r+4;" "
1510 IF INKEY$="w" OR INKEY$="W"
 THEN LET r=r+1: PRINT AT t,r;a$
;AT t+1,r;b$;AT t+2,r;c$: PRINT
AT t+2,r;d$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;c$
1520 IF INKEY$="q" OR INKEY$="Q"
 THEN LET r=r-1: PRINT AT t,r;e$
;AT t+1,r;f$;AT t+2,r;h$: PRINT
AT t+2,r;g$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;h$
1525 IF t=7 AND INKEY$="e" OR IN
KEY$="E" THEN LET t=4: PRINT ;AT
 t+4,r+1;"   ";AT t+3,r+1;"   ";
AT t+5,r;"   ";AT t,r;a$;AT t+1
,r;b$;AT t+2,r;c$: FOR b=0 TO 1:
 NEXT b: PRINT AT t,r;"   ";AT t
+1,r;"   ";AT t+2,r;"   ": LET
t=7
1530 IF r=14 AND t=2 THEN PRINT
AT 2,14;"    ";AT 3,14;"    ";AT
 4,14;"    ": LET t=7
1540 IF r=15 AND t=7 THEN PRINT
AT 7,14;"    ";AT 8,14;"    ";
AT 9,14;"    ": LET t=15
1550 IF t=7 AND r=y AND NOT INKE
Y$="e" THEN PRINT INK 3; FLASH 1
;AT t,r;"   ";AT t+1,r;"   ";A
T t+2,r;"   ": PRINT INK 3;AT 3
,19;"I";AT 4,7;"I";AT 8,3;"I";AT
 8,12;"T": FOR j=0 TO 40: BEEP .
035,j*RND: NEXT j: FLASH 0: PRIN
T FLASH 0;AT t,r;"   "; FLASH
0;AT t+1,r;"   "; FLASH 0;AT t
+2,r;"   ": LET t=2: LET r=22:
 LET f=f-1: PRINT AT 0,6;f
1560 IF t=15 THEN LET t=15
1570 IF t=2 AND r=19 AND INKEY$=
"q" THEN LET c=c+100: PRINT INK
6;AT 0,18;c
1580 IF t=7 AND r=2 OR r=11 AND
INKEY$="q" THEN LET c=c+100: PRI
NT AT 0,18;c
1582 IF t=15 AND r=13 AND INKEY$
="q" THEN LET c=c+100: PRINT AT
0,18;c
1583 IF t=7 AND r=7 AND INKEY$="
E" THEN LET c=c+100: PRINT AT 0,
18;c
1584 IF t=15 AND r=12 AND INKEY$
="q" THEN LET c=c+10
1586 IF t=15 AND r=13 THEN CLS :
 GO TO 2000
1590 IF t=15 AND r=23 THEN PRINT
 AT 16,26;"U": PRINT AT 16,26;"
": LET f=f-1: PRINT AT 0,6;f: PR
INT AT t,r;"   ";AT t+1,r+1;"
";AT t+2,r+1;"   ": LET t=2: LET
r=22
1600 IF f=0 THEN CLS : GO TO 300
1620 IF t=7 AND r=1 THEN LET r=r
+1: PRINT AT t+1,r;" "
1800 GO TO 1450
2000 REM
2100 LET a$=" BC ": LET b$=" DAE
 ": LET c$=" QP ": LET d$=" QR "
: LET e$="  IH ": LET f$=" JFG "
: LET g$="  LK ": LET h$="  MR "
2200 PRINT PAPER 6; INK 1;AT 10,
0;"█SSSSSSSSS█"
2201 PRINT PAPER 6; INK 1;AT 10,
17;"█SSSSSSSSSSSS█"
2202 PRINT INK 4;AT 17,0;"UUUUUU
UUUUUUUUUUUUUUUUUUUUUUUUUU"
2204 PRINT PAPER 6; INK 1;AT 18,
0;"█SSSSSSSSSSSSSSSSSSSSSSSSSSSSS
SS█"
2205 PRINT PAPER 6; INK 1;AT 19,
0;"█SSSSSSSSSSSSSSSSSSSSSSSSSSSSS
SS█"
2206 PRINT PAPER 3; INK 6;AT 20,
0;"█SSSSSSSSSSSSSSSSSSSSSSSSSSSSS
SS█"
2220 PRINT INK 3;AT 15,5;"I"
2221 PRINT INK 3;AT 15,25;"I"
2222 PRINT INK 3;AT 8,22;"I"
2300 PRINT INK 3;AT 0,0;"[redacted]"
2310 PRINT INK 5;AT 0,0;"lifes:"
;f
2320 PRINT INK 5;AT 0,10;"SCORE:
 ";c
2340 PRINT AT 3,25;"salida": PLO
T 196,143: DRAW 0,10: DRAW 55,0:
 DRAW 0,-14: DRAW -55,0: DRAW 0,
4
2400 LET t=14: LET r=10
2410 LET s=7: LET e=28
2420 LET a=16: LET l=29
3000 INK 5: PRINT INK 7;AT s,e;"
█";AT s,e+1;" "; INK 2;AT a,l;"U
";AT a,l+1;" ": LET e=e-1: LET l
=l-1: IF INKEY$="" THEN PRINT AT
 t,r;a$;AT t+1,r;b$;AT t+2,r;c$
3002 IF l=1 THEN LET l=l+1: PRIN
T AT 16,2;" ": LET l=29
3100 IF INKEY$="w" OR INKEY$="W"
 THEN LET r=r+1: PRINT AT t,r;a$
;AT t+1,r;b$;AT t+2,r;c$: PRINT
AT t+2,r;d$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;c$
3120 IF INKEY$="q" OR INKEY$="Q"
 THEN LET r=r-1: PRINT AT t,r;e$
;AT t+1,r;f$;AT t+2,r;h$: PRINT
AT t+2,r;g$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;h$
3140 IF t=14 AND INKEY$="e" OR I
NKEY$="E" THEN LET t=11: PRINT ;
AT t+4,r+1;"   ";AT t+3,r+1;"
 ";AT t+5,r;"   ";AT t,r;a$;AT
t+1,r;b$;AT t+2,r;c$: FOR b=0 TO
 1: NEXT b: PRINT AT t,r;"   ";A
T t+1,r;"   ";AT t+2,r;"   ": L
ET t=14
3150 IF t=14 AND r=13 AND INKEY$
="e" OR INKEY$="E" THEN LET t=7:
 PRINT ;AT t+7,r+1;"   ";AT t+8,
r+1;"   ";AT t+9,r;"   ";AT t,
r;a$;AT t+1,r;b$;AT t+2,r;c$: LE
T t=7
3151 IF t=14 AND r=12 AND INKEY$
="e" OR INKEY$="E" THEN LET t=7:
 PRINT ;AT t+7,r+1;"   ";AT t+8,
r+1;"   ";AT t+9,r;"   ";AT t,
r;a$;AT t+1,r;b$;AT t+2,r;c$: LE
T t=7
3152 IF t=14 AND r=11 AND INKEY$
="e" OR INKEY$="E" THEN LET t=7:
 PRINT ;AT t+7,r+1;"   ";AT t+8,
r+1;"   ";AT t+9,r;"   ";AT t,
r;a$;AT t+1,r;b$;AT t+2,r;c$: LE
T t=7
3155 IF t=7 AND INKEY$="e" OR IN
KEY$="E" THEN LET t=4: PRINT ;AT
 t+3,r+1;"   ";AT t+4,r+1;"   "
;AT t+5,r;"   ";AT t,r;a$;AT t+
1,r;b$;AT t+2,r;c$: FOR u=0 TO 1
: NEXT u: PRINT AT t,r;"   ";AT
 t+1,r;"   ";AT t+2,r;"   ": LE
T t=7
3160 IF t=7 THEN PRINT PAPER 6;
INK 1;AT 10,11;"SSSSSSS"
3200 IF e=1 THEN LET e=e+1: PRIN
T AT 7,2;" ": LET e=28
4000 IF t=14 AND r=l AND NOT INK
EY$="e" THEN PRINT INK 3; FLASH
1;AT t,r;"   ";AT t+1,r;"   ";
AT t+2,r;"   ": PRINT INK 3;AT
15,5;"I";AT 15,25;"I";AT 8,22;"I
": FOR j=0 TO 40: BEEP .035,j*RN
D: NEXT j: FLASH 0: PRINT FLASH
0;AT t,r;"   "; FLASH 0;AT t+1
,r;"   "; FLASH 0;AT t+2,r;"
   ": LET t=14: LET r=12: LET f=
f-1: PRINT AT 0,6;f
4001 IF t=7 AND r=e AND NOT INKE
Y$="e" THEN PRINT INK 3; FLASH 1
;AT t,r;"   ";AT t+1,r;"   ";A
T t+2,r;"   ": PRINT INK 3;AT 1
5,5;"I";AT 15,25;"T";AT 8,22;"T"
: FOR j=0 TO 40: BEEP .035,j*RND
: NEXT j: FLASH 0: PRINT FLASH 0
;AT t,r;"   "; FLASH 0;AT t+1,
r;"   "; FLASH 0;AT t+2,r;"
   ": LET t=14: LET r=12: LET f=f
-1: PRINT AT 0,6;f
4020 IF t=14 AND r=5 AND INKEY$=
"q" THEN LET c=c+100: PRINT AT 0
,18;c:
4021 IF t=14 AND r=22 AND INKEY$
="w" THEN LET c=c+100: PRINT AT
0,18;c:
4022 IF t=7 AND r=19 AND INKEY$=
"w" THEN LET c=c+100: PRINT AT 0
,18;c
4023 IF t=7 AND r=19 THEN CLS :
GO TO 5000
4100 IF f=0 THEN CLS : GO TO 300
4500 GO TO 3000
5000 REM 6 parte
5100 CLS
5104 LET v=6: LET n=22
5110 LET a$=" BC ": LET b$=" DAE
 ": LET c$=" QP ": LET d$=" QR "
: LET e$="  IH ": LET f$=" JFG "
: LET g$="  LK ": LET h$="  MR "
5120 LET t=17: LET r=5
5200 PRINT INK 3;AT 0,0;"[redacted]"
5400 PRINT INK 5;AT 0,0;"lifes:"
;f
5410 PRINT INK 5;AT 0,10;"SCORE:
 ";c
5500 PLOT INK 4;0,14: DRAW INK 4
;255,0: DRAW INK 4;0,-1: DRAW IN
K 4;-255,0
5501 INK 3: PLOT 160,16: DRAW 0
```

```
20: DRAW -3,0: DRAW 0,2: DRAW -3
,9: DRAW 0,7: DRAW 95,0
5502 PLOT 0,130: DRAW 12,-3: DRA
W -2,-7: DRAW 2,7: DRAW 2,6: DRA
W 16,-9: DRAW 15,7: DRAW 22,-7:
DRAW -4,-7: DRAW 4,7: DRAW 2,6:
DRAW 2,0: DRAW 12,-3
5503 DRAW 3,9: DRAW 24,-13: DRAW
 -3,0: DRAW 5,0: DRAW 2,7: DRAW
2,0: DRAW 2,9: DRAW 7,-12
5504 DRAW 2,0: DRAW 12,-6
5510 PLOT 0,80: DRAW 155,0: DRAW
 0,8
5511 PLOT 149,90: DRAW -5,9: DRA
W 29,0: DRAW 0,-10: DRAW -23,0
5512 PLOT 144,100: DRAW 0,5: DRA
W 3,0: DRAW 0,12: DRAW -3,0: DRA
W 0,6: DRAW 29,0: DRAW 0,-30
5513 PLOT 144,124: DRAW 24,24: D
RAW 5,0: DRAW 0,-37
5520 PLOT 180,88: DRAW 0,-3: DRA
W 5,0: DRAW 0,3
5530 PRINT INK 3;AT 18,3;"T"
5531 PRINT INK 3;AT 13,25;"T"
5540 PRINT AT 9,25;"salida": PLO
T 196,96: DRAW 0,10: DRAW 55,0:
DRAW 0,-14: DRAW -55,0: DRAW 0,4
5590 INK 5: PLOT 214,20: DRAW 0,
10: DRAW 3,0: DRAW 0,-10: PLOT 1
95,30: DRAW 0,12: DRAW 45,0: DRA
W 0,-12: DRAW -45,0: PRINT AT 17
,25;"out !"
5600 PRINT INK 3;AT v-1,n;" ";AT
 v,n;"█";AT v+1,n;"█": LET v=v+1
: INK 5: IF INKEY$="" THEN PRINT
 AT t,r;a$;AT t+1,r;b$;AT t+2,r;
c$
5620 IF r=0 THEN LET r=r+1: PRIN
T AT t+1,r;" "
5630 IF v=14 THEN LET v=v-1: PRI
NT AT 13,22;" ";AT 14,22;" ": LE
T v=8
5634 IF t=12 AND r=20 AND t=v AN
D NOT INKEY$="w" THEN PRINT AT t
,r;"    ";AT t+1,r;"    ";AT t+2,
r;"    ": LET t=17: LET r=5: LET
f=f-1: PRINT AT 0,6;f
5635 IF t=12 AND r=21 AND t=v AN
D NOT INKEY$="w" THEN PRINT AT t
,r;"    ";AT t+1,r;"    ";AT t+
2,r;"    ": LET t=17: LET r=5: L
ET f=f-1: PRINT AT 0,6;f
5640 IF INKEY$="q" OR INKEY$="Q"
 THEN LET r=r-1: PRINT AT t,r;e$
;AT t+1,r;f$;AT t+2,r;h$: PRINT
AT t+2,r;g$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;h$
5650 IF INKEY$="w" OR INKEY$="W"
 THEN LET r=r+1: PRINT AT t,r;a$
;AT t+1,r;b$;AT t+2,r;c$: PRINT
AT t+2,r;d$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;c$
5655 IF t=17 AND r=15 THEN LET r
=r-1: PRINT AT t+1,r+4;" "
5656 IF t=12 AND r=13 THEN LET r
=r+1: PRINT AT t+1,r;" "

5710 IF t=17 AND r=14 AND INKEY$
="e" OR INKEY$="E" THEN LET t=12
: PRINT ;AT t+5,r;"    ";AT t+6
,r;"    ";AT t+7,r;"    ";AT t,r
;a$;AT t+1,r;b$;AT t+2,r;c$: FOR
 b=0 TO 1: NEXT b: PRINT AT t,r;
"    ";AT t+1,r;"    ";AT t+2,r;"
    "
5800 IF t=17 AND r=3 AND INKEY$=
"q" THEN LET c=c+100: PRINT AT 0
,18;c
5801 IF t=12 AND r=22 AND INKEY$
="w" THEN LET c=c+100: PRINT AT
0,18;c
5900 IF r=23 THEN CLS : GO TO 61
00
5920 IF f=0 THEN CLS : GO TO 300
6020 GO TO 5600
6100 REM 6
6105 LET t=17: LET r=15
6200 LET a$=" BC ": LET b$=" DAE
 ": LET c$=" QP ": LET d$=" QR "
: LET e$=" IH ": LET f$=" JFG "
: LET g$=" LK ": LET h$=" MR "
6210 PRINT INK 3;AT 0,0;"[redacted]"
6220 PRINT INK 5;AT 0,0;"lifes:"
;f
6230 PRINT INK 5;AT 0,10;"SCORE:
 ";c
6400 INK 3: PLOT 20,10: DRAW 230
,0: DRAW -30,130: DRAW -14,9: DR
AW -14,3: DRAW -115,0: DRAW -14,
-3: DRAW -14,-10: DRAW -30,-130
6401 PLOT 18,10: DRAW 230,0: DRA
W -30,130: DRAW -14,9: DRAW -14,
3: DRAW -115,0: DRAW -14,-3: DRA
W -14,-10: DRAW -30,-130
6410 PLOT 38,90: DRAW 10,40: DRA
W 172,0: DRAW 10,-40: DRAW -190,
0
6420 PLOT 65,90: DRAW 3,12: DRAW
 3,2: DRAW 6,0: DRAW 3,-2: DRAW
1,-12
6421 PLOT 63,90: DRAW 3,14: DRAW
 3,2: DRAW 10,0: DRAW 3,-2: DRAW
 1,-12
6422 CIRCLE 74,96,3
6430 PLOT 40,49: DRAW 5,20: DRAW
 175,0: DRAW 5,-20: DRAW -9,12:
DRAW -167,0: DRAW -9,-12
6432 PRINT INK 3;AT 18,24;"T"
6434 PRINT AT 9,22;"salida": PLO
T 166,96: DRAW 0,10: DRAW 55,0:
DRAW 0,-14: DRAW -55,0: DRAW 0,4
6440 LET d=15: LET u=20
6490 PRINT INK 3;AT d,u;"U";AT d
-1,u;" ": LET d=d+1: INK 5: IF I
NKEY$="" THEN PRINT AT t,r;a$;AT
 t+1,r;b$;AT t+2,r;c$
6500 IF INKEY$="q" OR INKEY$="Q"
 THEN LET r=r-1: PRINT AT t,r;e$
;AT t+1,r;f$;AT t+2,r;h$: PRINT
AT t+2,r;g$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;h$
6502 IF d=19 THEN LET d=d-1: PRI
NT AT d,u;" ": PRINT AT d,u+RND*
20-12: LET d=15
6510 IF INKEY$="w" OR INKEY$="W"
 THEN LET r=r+1: PRINT AT t,r;a$
;AT t+1,r;b$;AT t+2,r;c$: PRINT
AT t+2,r;d$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;c$
6600 IF r=4 THEN LET r=r+1: PRIN
T AT t+1,r;" "
6601 IF r=25 THEN LET r=r-1: PRI
NT AT t+1,r+4;" "
6700 IF t=17 AND t=d AND r=19 TH
EN PRINT AT t,r;"    ";AT t+1,r;"
    ";AT t+2,r;"    ": LET t=17:
 LET r=5: LET f=f-1: PRINT AT 0,
6;f
6701 IF t=17 AND t=d AND r=18 TH
EN PRINT AT t,r;"    ";AT t+1,r;"
    ";AT t+2,r;"    ": LET t=17:
 LET r=5: LET f=f-1: PRINT AT 0,
6;f
6750 IF f=0 THEN CLS : GO TO 300
6800 IF r=20 AND INKEY$="w" THEN
 LET c=c+100: PRINT AT 0,18;c: C
LS : GO TO 7200
7000 GO TO 6490
7200 REM 7
7210 LET t=18: LET r=5
7221 LET a$=" BC ": LET b$=" DAE
 ": LET c$=" QP ": LET d$=" QR "
: LET e$=" IH ": LET f$=" JFG "
: LET g$=" LK ": LET h$=" MR "
7400 FOR m=0 TO 10 STEP 10: CIRC
LE INK 3;200,69,m: NEXT m
7410 PRINT INK 1; PAPER 5;AT 21,
0;"█SSSSSSSSSSSSSSSSSSSSSSSSSSSSS
SS[redacted]"
7411 PRINT INK 3;AT 0,0;"[redacted]"
7412 PRINT INK 5;AT 0,0;"lifes:"
;f
7415 PRINT INK 5;AT 0,10;"SCORE:
 ";c
7420 INK 4: CIRCLE 194,40,2: CIR
CLE 193,40,4: CIRCLE 194,24,2: C
IRCLE 193,24,4: INK 3: PLOT 170,
8: DRAW 0,47: DRAW 50,0: DRAW 0,
-47: DRAW 0,23: DRAW -50,0
7421 INK 3: PLOT 0,50: DRAW 170,
0: DRAW 0,5: DRAW -170,0: DRAW 0
,-5
7430 INK 6: PLOT 30,99: DRAW -2,
0: DRAW -4,9: DRAW 40,0: DRAW -4
,-9: DRAW -2,0: DRAW 3,7: DRAW -
34,0: DRAW 3,-7
7431 INK 6: PLOT 24,108: DRAW 0,
30: DRAW 40,0: DRAW 0,-30: DRAW
0,15: DRAW -40,0
7432 INK 6: PLOT 75,99: DRAW -2,
0: DRAW -4,9: DRAW 40,0: DRAW -4
,-9: DRAW -2,0: DRAW 3,7: DRAW -
34,0: DRAW 3,-7
7434 INK 6: PLOT 69,108: DRAW 0,
30: DRAW 40,0: DRAW 0,-30: DRAW
0,15: DRAW -40,0
7440 INK 6: PLOT 125,99: DRAW -2
,0: DRAW -4,9: DRAW 86,0: DRAW -
4,-9: DRAW -2,0: DRAW 3,7: DRAW
-40,0: DRAW 0,-7: DRAW -2,0: DRA
W 0,7: DRAW -40,0: DRAW 3,-7
7441 INK 6: PLOT 119,108: DRAW 0
,30: DRAW 86,0: DRAW 0,-30: DRAW
 0,15: DRAW -86,0
7450 PRINT INK 3;AT 19,3;"T";AT
13,8;"T"
7460 INK 5: LET x=20: LET y=13
7461 LET s=14: LET p=13
7490 PRINT INK 2;AT s,p;"UUUUU";
AT s,p+5;" ": LET p=p-1: PRINT I
NK 2;AT x,y;"UUUUU";AT x,y+5;" "
: LET y=y-1: INK 5: IF INKEY$=""
 THEN PRINT AT t,r;a$;AT t+1,r;b
$;AT t+2,r;c$
7492 IF y=1 THEN LET y=y+1: PRIN
T AT 20,2;"      ": LET y=14
7493 IF p=2 THEN LET p=p+1: PRIN
T AT 14,2;"      ": LET p=14
7500 IF INKEY$="w" OR INKEY$="W"
 THEN LET r=r+1: PRINT AT t,r;a$
;AT t+1,r;b$;AT t+2,r;c$: PRINT
AT t+2,r;d$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;c$
7510 IF INKEY$="q" OR INKEY$="Q"
 THEN LET r=r-1: PRINT AT t,r;e$
;AT t+1,r;f$;AT t+2,r;h$: PRINT
AT t+2,r;g$: FOR b=0 TO 2: NEXT
b: PRINT AT t+2,r;h$
7520 IF t=18 AND INKEY$="e" OR I
NKEY$="E" THEN LET t=16: PRINT ;
AT t+3,r+1;"    ";AT t+4,r+1;"
   ";AT t+3,r;"    ";AT t,r;a$;AT
 t+1,r;b$;AT t+2,r;c$: FOR u=0 T
O 1: NEXT u: PRINT AT t,r;"    "
;AT t+1,r;"    ";AT t+2,r;"    ":
 LET t=18
7530 IF t=18 AND r=15 AND INKEY$
="e" OR INKEY$="E" THEN LET t=12
: PRINT ;AT t+6,r+1;"    ";AT t+
7,r+1;"    ";AT t+8,r;"    ";AT
t,r;a$;AT t+1,r;b$;AT t+2,r;c$:
FOR u=0 TO 1: NEXT u: PRINT AT t
,r;"    ";AT t+1,r;"    ";AT t+2
,r;"    "
7531 IF t=12 AND INKEY$="e" OR I
NKEY$="E" THEN LET t=10: PRINT ;
AT t+3,r+1;"    ";AT t+4,r+1;"
  ";AT t,r;a$;AT t+1,r;b$;AT t+
2,r;c$: FOR a=0 TO 1: NEXT a: PR
INT AT t,r;"    ";AT t+1,r;"
";AT t+2,r;"    ": LET t=12
7540 IF t=18 AND r=p AND NOT INK
EY$="e" THEN PRINT INK 3;AT t,r;
"    ";AT t+1,r;"    ";AT t+2,r;
"    ": BEEP .035,-10: LET r=6:
LET f=f-1: PRINT AT 0,6;f: PRINT
 INK 3;AT 19,3;"T";AT 13,8;"T"
7541 IF t=12 AND r=y AND NOT INK
EY$="e" THEN PRINT INK 3;AT t,r;
"    ";AT t+1,r;"    ";AT t+2,r;
"    ": BEEP .035,-10: LET r=8:
LET f=f-1: PRINT AT 0,6;f: PRINT
 INK 3;AT 19,3;"T";AT 13,8;"T"
7600 IF r=1 THEN LET r=r+1: PRIN
T AT t+1,r;" "
7601 IF t=18 AND r=16 THEN LET r
=r-1: PRINT AT t+1,r+4;" "
7800 IF t=12 AND r=20 AND INKEY$
="w" THEN LET c=c+500: PRINT AT
0,18;c
7812 IF t=18 AND INKEY$="q" AND
r=3 THEN LET c=c+100: PRINT AT 0
,18;c
7814 IF t=12 AND INKEY$="q" AND
r=8 THEN LET c=c+100: PRINT AT 0
,18;c
7820 IF r=24 THEN CLS : GO TO 82
00
7900 IF f=0 THEN CLS : GO TO 300
8000 GO TO 7490
8200 CLS
8300 PRINT AT 2,6;"HALL OF THE F
AME": PLOT 47,150: DRAW 128,0
8400 PRINT AT 5,7;"W.S.B....800"
;AT 7,7;"EUGENIUDS..600";AT 9,7;
"E.S.P...300"
8410 IF c>300 THEN PRINT AT 12,1
;"Estas entre los mejores."
8411 IF c>300 THEN INPUT "pon tu
s iniciales";a$
8420 IF c>300 OR c>320 THEN PRIN
T AT 9,7;a$;"........";c;" "
8421 IF c>600 OR c>620 THEN PRIN
T AT 7,7;a$;"........";c;" "
8422 IF c>800 OR c>820 THEN PRIN
T AT 9,7;a$;"........";c;" "
8500 IF c>1000 THEN PRINT FLASH
1; PAPER 6; INK 1;AT 5,7;a$; FLA
SH 0;"........";c;" "
8880 PRINT AT 20,2;"Pulse cualqu
ier tecla": PAUSE 0
8890 FOR g=0 TO 55: BEEP .025,g+
INT (RND+0.035): NEXT g: CLS : G
O TO 300
```

Programas

44



Para Elisa

Recientemente asistimos a una proliferación de programas de música, especialmente de la casa Musi-Soft e Idealogic, afincadas en Madrid y Barcelona respectivamente. En próximos números de TODOSPECTRUM analizaremos en detalle las posibilidades "serias" en cuanto a música y Spectrum se refiere. Y aprovechamos para aclarar un error ya clásico: El Spectrum es tan bueno para programas musicales como cualquier otro. Las posibilidades gráficas para dibujo de "la partitura" son si cabe más importante que el sonido.

Por el momento, y para "abrir boca" comenzaremos con un pequeño programa: el célebre PARA ELISA de Beethoven. Un corto programa, cuya extensión se podría haber reducido considerablemente con la utilización de bucles y datos en instrucciones READ-DATA. En cualquier caso, lo importante es el sonido y la posibilidad de incorporarlo a otros programas.

Autor: Francisco Martínez *16K*

```
 380 REM ©FRANCISCO MARTINEZ
 390 BORDER 0: PAPER 0: INK 7: C
LS
 392 PRINT AT 5,12; FLASH 1;"PAR
A ELISA";AT 7,8; FLASH 0;"de L.V
an Beethoven";AT 15,2;"Adaptado
al zx-Spectrum por     Francisco
Martinez Morenilla"
 395 LET t=1.2: LET n0=0: LET n1
=2: LET n2=4: LET n3=5: LET n4=7
: LET n5=9: LET n6=11: LET n7=12
: LET n8=14: LET n9=16: LET n10=
17
 400 LET n11=19: LET n12=21: LET
 n13=23: LET n14=24: LET n15=25:
 LET n16=8: LET n17=10: LET n18=
16: LET n19=23
 500 LET a=.0625*t: LET b=.125*t
: LET c=.25*t: LET d=.5*t: LET e
=1*t: LET bb=.1875*t: LET cc=.37
5*t
 505 LET cp=c*30: LET dp=d*30
 510 GO SUB 800: BEEP cc,n5: PAU
SE cp: GO SUB 800: BEEP c,n5: PA
USE cp: BEEP b,n6: BEEP b,n7: BE
EP b,n8: GO SUB 700
 520 BEEP c,n5: PAUSE cp: BEEP b
,n6: BEEP b,n7: BEEP b,n8: GO SU
B 700:: BEEP c,n6: PAUSE cp: BEE
P b,n7: BEEP b,n7: BEEP b,n7: BE
EP d,n7: BEEP cc,n10: BEEP a,n9
 540 BEEP c,n9: BEEP c,n8: BEEP
bb,n19: BEEP a,n12: BEEP a,n12:
BEEP b,n11: BEEP b,n10: BEEP b,n
9: BEEP b,n8: BEEP b,n7: BEEP c,
n17: BEEP c,n5: BEEP a,n5
 560 BEEP a,n4: BEEP a,n5: BEEP
a,n17: BEEP d,n7: BEEP b,n8: BEE
P b,n18: BEEP cc,n9: BEEP b,n9:
BEEP b,n10: BEEP b,n5: BEEP d,n7
: BEEP bb,n8: BEEP a,n6
 580 BEEP a,n7: BEEP a,n11: BEEP
 a,n4: BEEP a,n11: BEEP a,n6: BE
EP a,n11: BEEP a,n5: BEEP a,n11:
 BEEP a,n7: BEEP a,n11: BEEP a,n
8
 582 BEEP a,n11: BEEP a,n9: BEEP
 a,n11: BEEP a,n14: BEEP a,n13:
BEEP a,n12: BEEP a,n11: BEEP a,n
10: BEEP a,n9: BEEP a,n8: BEEP a
,n11: BEEP a,n10: BEEP a,n8: BEE
P a,n7: BEEP a,n11: BEEP a,n4: B
EEP a,n11: BEEP a,n5: BEEP a,n11
: BEEP a,n6: BEEP a,n11: BEEP a,
n7: BEEP a,n11: BEEP a,n8: BEEP
a,n11: BEEP a,n9
 584 BEEP a,n11: BEEP a,n14: BEE
P a,n13: BEEP a,n12: BEEP a,n11:
```

```
 BEEP a,n10: BEEP a,n9: BEEP a,n
8: BEEP a,n11: BEEP a,n10: BEEP
a,n8: BEEP a,n9: BEEP a,n10: BEE
P a,n9: BEEP a,n8: BEEP a,n9: BE
EP a,n6: BEEP a,n9: BEEP a,n18:
BEEP a,n9: BEEP a,n6: BEEP a,n8:
 BEEP a,n9: BEEP a,n18: BEEP cc,
n9: BEEP b,n6
 586 BEEP b,n9: BEEP b,n18: BEEP
 cc,n9: BEEP b,n8: BEEP b,n9: BE
EP b,n18: BEEP b,n9: BEEP b,n18:
 BEEP b,n9: BEEP b,n18: GO SUB 8
00: BEEP c,n5: PAUSE dp: BEEP b,
n6: BEEP b,n7: BEEP b,n8: GO SUB
 700: BEEP e,n5: GO TO 1000
 700 BEEP cc,n9: BEEP b,n4: BEEP
 b,n10: BEEP b,n9: BEEP cc,n8: B
EEP b,n3: BEEP b,n9: BEEP b,n8:
BEEP cc,n7
 710 BEEP b,n2: BEEP b,n8: BEEP
b,n7: BEEP b,n6: BEEP b,n2: BEEP
 b,n9: BEEP b,n2: BEEP b,n9: BEE
P b,n2: BEEP b,n9: BEEP b,n9: BE
EP b,n15: BEEP b,n18: BEEP b,n9
 720 BEEP b,n18: BEEP b,n9: BEEP
 b,n18: BEEP b,n9: BEEP b,n18: G
O SUB 900: BEEP b,n2: BEEP b,n16
: BEEP b,n6: BEEP c,n7: PAUSE cp
: BEEP b,n2: GO SUB 900: BEEP b,
n1: BEEP b,n7: BEEP b,n6: RETURN

 800 GO SUB 900: BEEP b,n2: BEEP
 b,n16: BEEP b,n6: BEEP b,n7: PA
USE cp: BEEP b,n2: GO SUB 900: B
EEP b,n1: BEEP b,n7: BEEP b,n6:
RETURN
 900 BEEP b,n9: BEEP b,n18: BEEP
 b,n9: BEEP b,n18: BEEP b,n9: BE
EP b,n6: BEEP b,n8: BEEP b,n7: B
EEP b,n5: PAUSE cp: BEEP b,n0: B
EEP b,n2: BEEP b,n5: BEEP b,n6:
PAUSE cp: RETURN
```

PARA ELISA

de L.Van Beethoven

Adaptado al
zx-Spectrum por
Francisco Martinez
Morenilla

La cara loca

Programa especialmente pensado para los niños, con gran diversidad de caras que se formarán ante sus ojos, pudiendo participar con el teclado de una forma sencilla y entretenida.

Las caras que salen son completamente aleatorias, guardando todas unos esquemas previamente programados. Por ejemplo, el contorno de la cabeza, los ojos, las cejas, la nariz, la boca, son cosas que siempre salen, pero variando dentro de unos límites. La boca puede ser sonriente o triste o de labios gruesos o delgados.

También hay cosas que pueden o no salir, como son: pecas, pelo, un cigarro o pipa, gafas, etc., y cuando salen lo hacen de distinto tamaño o incluso posición en pantalla diferente.

La expresión utilizada más frecuentemente en el programa es:

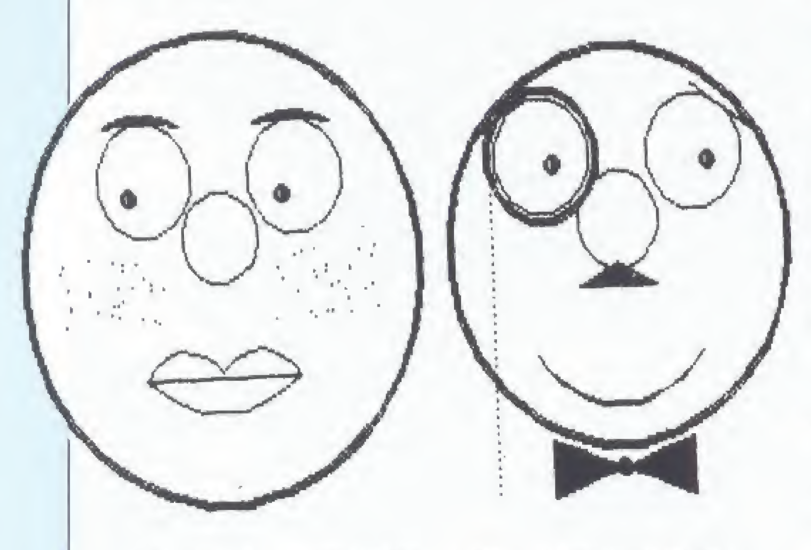
LET H = INT (RND * 7)
IF H = 4 THEN...
IF H > 4 THEN...

donde RND 7 genera un número aleatorio entre 0.0000001 y 6.999999 y INT redondea al entero, por lo que los números útiles van del 0 al 6. Si la expresión hubiese sido LET H = INT (5 + RND * 7) el número oscilaría entre 5 y 11. Esto luego puede ser aplicado al radio de un círculo o a una posición de pantalla, o lo que se quiera. Esto hace que las caras sean siempre diferentes y las que se parecen nunca son exactamente iguales.

También varían el color del BORDE y PAPEL manteniéndose el color de la tinta en contraste mediante INK 9, y además el BRILLO y el OVER.

El funcionamiento del programa es muy sencillo. En principio están

```
  10 LET REP=10: RANDOMIZE 0: P
OKE 23658,8
  20 BORDER 0: PAPER 0: INK 5: C
LS
  30 LET us=300: LET po=0: LET p
un=0
  40 LET yy=20: LET xs=5: LET ys
=5: LET p$="LA": GO SUB 1140
  50 LET yy=70: LET p$="CARA": G
O SUB 1140
```



```
  60 LET yy=120: LET p$="LOCA":
GO SUB 1140
  70 BEEP .2,5: PAUSE 5: BEEP .5
,5: BEEP 1,12: BEEP .5,5: PAUSE
5: BEEP .5,5: BEEP .5,14: BEEP 1
,12: PAUSE 50: CLS
  80 LET yy=0: LET xs=10: LET ys
=10: INK 6: LET p$=STR$ REP: GO
SUB 1140: LET yy=130: LET xs=2:
LET ys=3: INK 4: LET p$="CARAS":
 GO SUB 1140: FOR n=1 TO 200: NE
XT n
  90 PRINT ^1; FLASH 1;"PULSA UN
A TECLA": PAUSE 0
 100 FOR e=1 TO INT REP
 110 LET a=INT (RND*8)
 120 LET OV=INT (RND*5): IF OV T
HEN  LET OV=0: GO TO 140
 130 LET OV=1
 140 BORDER a: PAPER a: INK 9: B
RIGHT RND*1: OVER OV: CLS : RAND
OMIZE 0
 150 LET xx=200: LET yy=175: LET
 xs=2: LET ys=2: LET p$=STR$ e:
GO SUB 1150
 160 LET ca=INT (RND*5): IF NOT
ca THEN  LET ci=90: GO TO 190
 170 LET ci=INT (70+RND*18): FOR
 n=ci TO ci-2 STEP -1: CIRCLE 12
7,88,n: NEXT n
 180 GO TO 210
 190 LET cu=RND*3.2: LET po=INT
(10+RND*25): FOR N=CU TO CU-.1 S
TEP -.05: PLOT 127-po,175: DRAW
0,-175,N: PLOT 127+po,175: DRAW
0,-175,-N: NEXT N: PLOT 127-po,1
75: DRAW po*2,0: PLOT 127-po,0:
DRAW po*2,0
 200 LET OJ=INT (RND*10): IF OJ>
5 THEN  GO TO 220
```

programadas 10 caras (que nunca serán las mismas al ejecutar el programa por vez primera). Por cada cara que sale aparecerá una letra en la esquina inferior izquierda. Al pulsarse la tecla correspondiente se ejecutará otra cara. La letra o número que aparece estará un tiempo determinado en pantalla y si no se pulsa la tecla correcta aparecerá otra letra o número. Este tiempo viene determinado por la variable **us** y en principio tiene un valor de 300, disminuyendo en cada cara en 25. Si se desea se puede cambiar su valor inicial si el niño tiene dificultades con el teclado. Por ejemplo, LET us = 1000.

Una vez que han salido las 10 caras el niño se enfrenta a unas letras que aparecerán en pantalla de distinto tamaño y color, y deberá proceder como en las anteriores: pulsar la tecla adecuada.

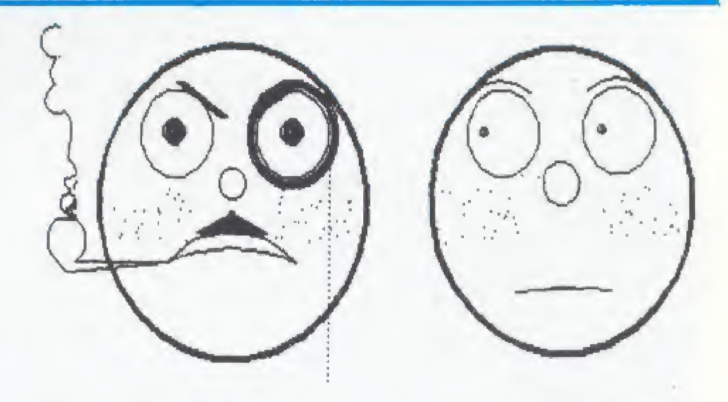
Al terminar esta pantalla saldrá la puntuación. La máxima puede ser de 20, por lo que el programa empezará otra vez con 10 caras. Si los puntos hubiesen sido de 8 se empezaría de nuevo con 4 caras.

Otra cosa que se incluye en el programa y que puede resultar interesante es una salida para impresora. Si se tiene conectada, la cara pasa a la impresora cuando el niño pulsa la tecla correcta. Estas caras, ya pasadas al papel, se pueden colorear o recortar o hacerse con un buen archivo para regalar a los amigos.

Para finalizar hay que decir que el programa se apoya en una subrutina en código máquina para generar caracteres gigantes. Esta subrutina es la misma que la que viene en la cinta HORIZONTES, cara B y se obtiene mediante LOAD "c" CODE.

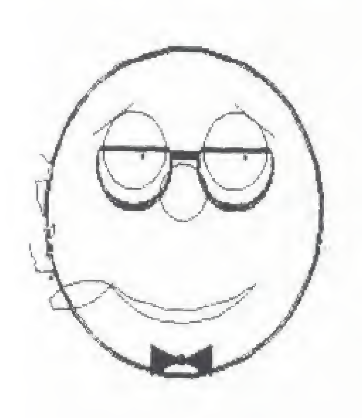
José C. Tomás

```
 210 CIRCLE 96,120,20: CIRCLE 15
9,120,20
 220 LET PN=INT (96+RND*10): LET
 TN=INT (5+RND*13): CIRCLE 127,P
N,TN
 230 LET O=INT (RND*7)
 240 IF O>4 THEN  GO SUB 650
 250 IF O<4 THEN  GO SUB 710
 260 IF O=4 THEN  LET PO=INT (10
-RND*20): PLOT 77,119: DRAW 38,0
: PLOT 140,119: DRAW 38,0: CIRCL
E 95+PO,115,3: CIRCLE 159+PO,115
,3
 270 GO SUB 500
 280 LET A=INT (RND*4)
 290 IF A=0 THEN  LET GR=INT (RN
D*10): FOR N=96 TO 100: PLOT N,1
44: DRAW 24+GR,-16: NEXT N
 300 IF A=0 THEN  FOR N=159 TO 1
63: PLOT N,144: DRAW -24-GR,-16:
 NEXT N
```





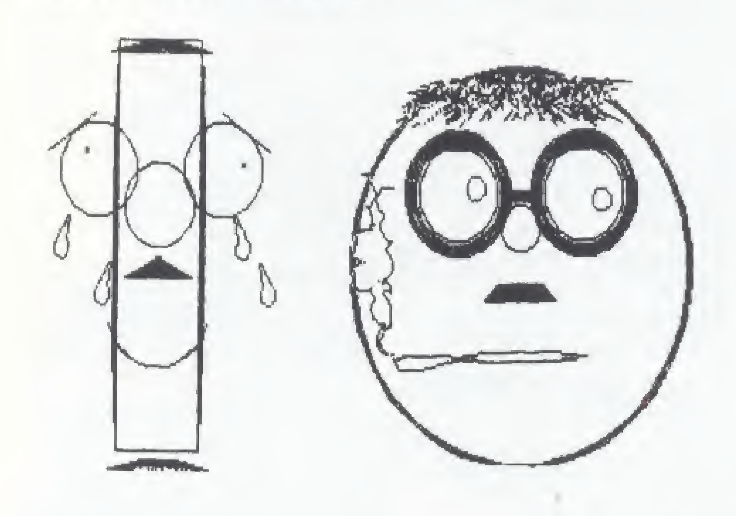
```
 310 IF A=1 THEN  LET CE=(RND*3)
: FOR N=CE TO CE-.6 STEP -.2: PL
OT 80,139: DRAW 32,0,-N: PLOT 14
3,139: DRAW 32,0,-N: NEXT N
 320 IF A=2 THEN  PLOT 95,144: D
RAW -24,-16: PLOT 159,144: DRAW
24,-16
 330 LET Q=INT (RND*6): IF Q=0 T
HEN  GO SUB 730
 340 LET A=INT (RND*10): IF A=5
THEN  GO SUB 770
 350 LET PE=INT (RND*5): IF PE=3
 THEN  GO SUB 890
 360 LET LO=INT (RND*5): IF LO>2
 THEN  GO SUB 930
 370 LET LA=INT (RND*10): IF LA=
3 THEN  GO SUB 1030
 380 LET JA=INT (RND*10): IF JA>
=7 THEN  GO SUB 1100
```





```
 390 LET ST=INT (1+RND*10): FOR
N=-20 TO 20 STEP ST: BEEP .09,N:
 BEEP .01,N+20: NEXT N
 400 IF pun<0 THEN  LET pun=0
 405 LET LE=INT (RND*10)
 410 IF LE THEN  LET A$=CHR$ INT
 (65+RND*26): GO TO 430
 420 LET A$=CHR$ INT (48+RND*10)
 430 FOR M=1 TO 4: LET xx=0: LET
 yy=160: LET xs=M: LET ys=M: LET
 p$=A$: GO SUB 1150: NEXT M
 440 FOR n=1 TO us: IF INKEY$<>a
$ THEN  NEXT n
 450 IF INKEY$<>a$ THEN  BEEP .5
,-20: LET p$=" ": GO SUB 1150: L
ET pun=pun-1: GO TO 400
 460 FOR n=1 TO 60 STEP 3: BEEP
.01,n: NEXT n
 470 LET xx=1: LET yy=160: LET x
s=5: LET ys=5: LET p$=" ": GO SU
B 1150
 480 IF INKEY$<>"" THEN  GO TO 4
60
 490 COPY : LET us=us-25: LET pu
n=pun+1: NEXT e: GO TO 1160
 500 LET U=INT (80+RND*40): LET
D=(127-U)*2: LET J=50: LET I=(RN
D*2)
 510 LET F=INT (RND*4): IF F=0 T
HEN  PLOT U,J: DRAW D,0,I+D/500:
 PLOT U,J: DRAW D,0,I+(D/500)+RN
D
 520 IF F=1 THEN  PLOT U,J: DRAW
 D,0,-I: PLOT U,J: DRAW D,0,-I-R
ND
 530 IF F=2 THEN  PLOT U,J: DRAW
 D,0,-I: PLOT U,J: DRAW D,0,I
 540 IF F=3 THEN  PLOT U,J: DRAW
 D/2,2,-I: DRAW D/2,-2,-I: PLOT
U,J: DRAW D,0: PLOT U,J: DRAW D,
0,I-.5
 550 LET RR=INT (RND*10): IF RR=
5 THEN  GO SUB 570
 560 RETURN
 570 LET PI=INT (RND*2): IF PI T
HEN  GO TO 620
 580 LET CC=RND*1: LET MO=-2: LE
T LA=INT (15+RND*65): LET GA=INT
 (RND*10)
 580 LET CC=RND*1: LET MO=-2: LE
T LA=INT (15+RND*65): LET GA=INT
 (RND*10)
 590 PLOT U,J: DRAW -LA,-GA,CC:
DRAW 3,-5: DRAW LA,GA+5,CC: PLOT
 U-LA,J-MO
 600 LET HU=INT (1+RND*5)
 610 FOR M=1 TO HU: FOR N=1 TO R
ND*10: DRAW  OVER 1;-5+RND*10,RN
D*15,-4+RND*8: NEXT N: PLOT U-LA
+M,J-MO: NEXT M: RETURN
 620  PLOT U,J: DR  10 LET REP=
10: RANDOMIZE 0: POKE 23658,8
  20 BORDER 0: PAPER 0: INK 5: C
LS
  30 LET us=300: LET po=0: LET p
un=0
  40 LET yy=20: LET xs=5: LET ys
=5: LET p$="LA": GO SUB 1140
  50 LET yy=70: LET p$="CARA": G
 570>LET PI=INT (RND*2):IF PITHE
N GO TO 620
```

```
 580 LET CC=RND*1: LET MO=-2: LE
T LA=INT (15+RND*65): LET GA=INT
 (RND*10)
 590 PLOT U,J: DRAW -LA,-GA,CC:
DRAW 3,-5: DRAW LA,GA+5,CC: PLOT
 U-LA,J-MO
 600 LET HU=INT (1+RND*5)
 610 FOR M=1 TO HU: FOR N=1 TO R
ND*10: DRAW  OVER 1;-5+RND*10,RN
D*15,-4+RND*8: NEXT N: PLOT U-LA
+M,J-MO: NEXT M: RETURN
 620  PLOT U,J: DRAW -50,0: DRAW
 0,20,2: DRAW -10,0: DRAW 0,-20,
2: DRAW 5,-5: DRAW 55,5
 630 LET MO=-22: LET LA=54: PLOT
 U-LA,J-MO: GO TO 600
 640 LET OJ=INT (RND*2): IF OJ=1
 THEN  GO TO 680
 650 LET X=83+INT (RND*23): LET
Y=107+INT (RND*25)
 740 FOR M=74 TO 80+LO: PLOT 112
-BI+TE,M: DRAW 30+(BI*2)-(2*TE),
0
 750 IF 30+(BI*2)-(2*TE)<=0 THEN
 RETURN
 760 LET TE=TE+IG: NEXT M: RETUR
N
 770 LET GAF=INT (RND*10): IF GA
F=9 THEN  GO TO 820
 780 IF GAF=8 THEN  GO TO 840
 790 IF GAF=7 THEN  GO TO 860
 800 LET GA=INT (23+RND*6): FOR
N=22 TO GA: CIRCLE 96,120,N: CIR
CLE 159,120,N: NEXT N
 810 LET AF=(RND*3): FOR M=116 T
O 120: PLOT 119,M: DRAW 17,0,-AF
: NEXT M: RETURN
 820 LET GA=INT (23+RND*7): FOR
N=22 TO GA: CIRCLE 96,120,N: NEX
T N
```







```
 660 LET TO=INT (1+RND*6): FOR N
=1 TO TO: CIRCLE X,Y,N: CIRCLE 6
4+X,Y,N: NEXT N
 670 RETURN
 680 LET X=83+INT (RND*23): LET
Y=107+INT (RND*25)
 690 CIRCLE X,Y,N: CIRCLE 64+X,Y
,N
 700 RETURN
 710 LET TA=INT (1+RND*5): LET X
=INT (20+RND*25): LET Y=107+INT
(RND*25): CIRCLE 127+X,Y,TA: CIR
CLE 127-X,Y,TA
 720 RETURN
 730 LET TE=0: LET IG=INT (RND*5
): LET LO=INT (RND*10): LET BI=I
NT (RND*10)
 830 FOR n=110 TO 0 STEP -3: PLO
T 75,n: NEXT n: RETURN
 840 LET GA=INT (23+RND*7): FOR
N=22 TO GA: CIRCLE 159,120,N: NE
XT N
 850 FOR n=110 TO 0 STEP -3: PLO
T 180,n: NEXT n: RETURN
 860 FOR N=120 TO 122: PLOT 75,N
: DRAW 43,0: PLOT 138,N: DRAW 43
,0: NEXT N
 870 FOR N=4 TO 3.8 STEP -.05: P
LOT 75,122: DRAW 43,0,N: PLOT 13
8,122: DRAW 43,0,N: NEXT N
 880 GO TO 810
 890 IF CI=90 THEN  RETURN
 900 FOR N=1 TO INT (5+RND*25):
```

```
LET PE=INT (60+RND*45): LET CA=I
NT (70+RND*25): PLOT PE,CA
 910 LET PE=INT (60+RND*45): LET
 CA=INT (70+RND*25): PLOT PE+90,
CA
 920 NEXT N: RETURN
 930 LET OV=INT (RND*2): IF OV T
HEN  OVER 1: GO TO 950
 940 OVER 0
 950 LET LA=INT (RND*15)
 960 FOR N=20 TO INT (20+RND*100
)
 970 IF ci+90>=175 THEN  LET ci=
85
 980 LET PE1=INT (127-po+RND*po*
2)
 990 PLOT PE1,ci+90
1000 LET PE=INT (50-RND*100): LE
T LO=-INT (RND*20)
1010 DRAW PE,LO-LA: PLOT PE1,ci+
90: DRAW -PE,LO-LA
1020 NEXT N: OVER 0: RETURN
1030 IF F=0 THEN  RETURN
1040 LET GR=0
1050 OVER 1: LET RI=INT (RND*12)
: LET RI=11: LET LAG=INT (RND*50
): FOR N=100 TO 10+LAG STEP -20
1060 PLOT INT (80+RND*30)-GR,N:
DRAW -2,-RI: DRAW -5,0,-5: DRAW
6,RI
1070 PLOT INT (150+RND*30)+GR,N:
 DRAW 6,-RI: DRAW -5,0,-5: DRAW
-2,RI
1080 LET GR=GR+5: NEXT N
1090 OVER 0: RETURN
1100 LET TAM=INT (5+RND*6): LET
PA=INT (10+RND*21)
1110 FOR N=1 TO 3: CIRCLE 127,10
,N: NEXT N
1120 FOR N=TAM TO -TAM STEP -1:
PLOT 127,10: DRAW PA,N: PLOT 127
,10: DRAW -PA,N: NEXT N
1130 RETURN
1140 LET xx=(256-8*xs*LEN p$)/2
1150 LET i=23306: POKE i,xx: POK
E i+1,yy: POKE i+2,xs: POKE i+3,
ys: POKE i+4,8: LET i=i+4: LET w
=LEN p$: FOR n=1 TO w: POKE i+n,
CODE p$(n): NEXT n: POKE i+w+1,2
55: LET w=USR 32256: RETURN
1160 BORDER 0: PAPER 0: BRIGHT 1
: CLS
1170 FOR m=1 TO 10: LET xx=INT (
RND*180): LET xs=INT (1+RND*10):
 LET ys=xs: LET d$=CHR$ INT (65+
RND*26): INK INT (2+RND*6)
1180 LET yy=INT (RND*130): LET p
$=d$: GO SUB 1150
1190 FOR n=1 TO 100: IF INKEY$=d
$ THEN  FOR b=1 TO 60 STEP 3: BE
```

```
EP .01,b: NEXT b: CLS : LET pun=
pun+1: GO TO 1210
1200 NEXT n: BEEP .1,-20: CLS :
LET pun=pun-1: IF pun<0 THEN  LE
T pun=0
1210 NEXT m
1220 LET yy=10: LET xs=2: LET ys
=2: LET p$="PUNTOS": GO SUB 1140
: LET yy=100: LET xs=6: LET ys=6
: LET p$=STR$ pun: GO SUB 1140
1230 FOR N=1 TO 400: NEXT N: CLS
 : LET REP=INT (pun/2)
1240 IF pun>=20 THEN  GO TO 1270
1250 FOR M=1 TO 7: LET yy=60+(M*
10): LET xs=2: LET ys=3: INK M:
LET p$="OTRA PARTIDA DE": GO SUB
 1140: NEXT M: LET yy=88: LET xs
=10: LET ys=10: LET p$=STR$ REP:
 GO SUB 1140: PAUSE 300
1260 LET us=REP*30: GO TO 20
1270 OVER 1: FOR N=0 TO 175 STEP
 INT (1+RND*10): PLOT N,N: DRAW
255-(2*N),0: DRAW 0,175-(2*N): D
RAW -255+(2*N),0: DRAW 0,-175+(2
*N): NEXT N
1280 LET yy=83: LET ys=2: LET xs
=3: LET p$="MUY BIEN": GO SUB 11
40: FOR N=1 TO 300: NEXT N: GO T
O 20
1290 STOP
1300 CLEAR 32255: LOAD "b1"CODE
32256: RUN
9999 SAVE "CARA LOCA" LINE 1300:
 SAVE "b1"CODE 32256,300: VERIFY
 "": VERIFY ""CODE
```



Programas/Juegos/Montajes/Còdigo Màquina
AÑO II · Núm. 18 · Mayo 1985 · 250 Ptas.
ZX
REVISTA PARA LOS USUARIOS DE ORDENADORES SINCLAIR
Renta 85
listado para rellenar la declaración
Plotting gráficos
Ventanas móviles
YA ESTA A LA VENTA
¡GANE ESTA VESPA!
BANCO DE ESPAÑA
CINCO MIL pesetas
BANCO DE ESPAÑA
MIL pesetas
12.587.061.720
2.004.791362

Preguntas y respuestas

P En el artículo «Del 48 al plus. El kit paso a paso» (págs. 4, 5, 6, 7 y 8 de la revista n.º 7) dicen Vds. que las únicas diferencias entre el Spectrum y el Plus son: a) un botón de RESET, b) una caja más amplia que refrigera a través de dos rejillas de ventilación y c) un nuevo teclado que incorpora 17 teclas más. *¿Qué pasa con el supuesto aumento de memoria a 64K?* También dicen Vds. que este kit de montaje lo proporcionará Investrónica, *¿cuándo será esto y tendrá el mismo precio que en Inglaterra* (6.000 ptas sólo el kit y 4.000 más si lo montan ellos)? Porque en la pág. 2 de la revista ZX n.º 17 Investrónica anuncia un cambio de Spectrum a Plus por 12.000 ptas. *¿es esto lo que Vds. decían que haría Investrónica?*

Christian Klein
Barcelona

R *No se trata de un aumento de memoria a 64K. Sigue siendo el mismo Spectrum de 16K de ROM y 48K de RAM. Por razones de marketing se eligió la denominación de 64K, que tanta confusión ha generado. Igualmente confuso parece resultar el último anuncio de* **Investrónica** *del paso del Spectrum al Plus, que en realidad hace referencia al Kit del Plus que comentábamos en el número 7.*

P He realizado el montaje del lápiz óptico del número 6 pero no me funciona bien. ¿Se han encontrado con algún problema?

Juan C. Pérez
Segovia

R *Nos alegra tu pregunta que aprovechamos para hacer algunas matizaciones sobre dicho montaje:*

— En la lista de componentes aparece un condensador C_1 de 10 µF, cuando en realidad debe ser de 10 nF.

— Todas las resistencias son de 1/4.

— La descripción del apantallamiento del circuito no fue muy afortunada. En la figura 4 se puede ver que junto a TS existe un terminal marcado con el signo de masa. Ayúdese también en la figura 9. Suelde en ese orificio un hilo y elimine la mitad del plástico aislante del extremo opuesto al de la soldadura. Envuelva a continuación el circuito con plástico o papel y luego con papel de aluminio, pero dejando salir el mencionado conductor. Enrolle al papel de aluminio, con el doble objeto de proporcionar un buen contacto eléctrico y rigidez mecánica.

La localización del fotodiodo puede resultar difícil. Sin embargo, cualquier fotodiodo que funcione en el espectro del visible es válido. Un sustituto bastante conocido es el BPW66.

P El pasado día 2 adquirí interface "INDESCOMP-Centronics-RS-232" y una impresora GP-550 A de "SEIKOSHA". Cuando conecté los dos periféricos a mi "Spectrum 48K" me percaté de que no podía hacer "copys" de pantalla porque según explicáis en la revista número 2 de octubre, pág. 44, se necesita para ello un programa adicional que distribuye la empresa "DIRAC". Pues referente a esto último son mis preguntas que, abusando de vuestra amabilidad, espero me contestéis.

a) Cuando tenga el programa de "copy" y lo cargue en el ordenador, ¿podré cargar otro programa distinto y trabajar con él?

b) Cuando el programa "copy" esté cargado, ¿qué debo hacer para que todo lo que esté en pantalla salga copiado exactamente igual en la impresora?

c) Puesto que el cuadernillo de instrucciones del interface señala que los "copys" que pudieran estar en cualquier programa se deberán sustituir por los correspondientes "RANDOMIZE USR 65044 o 650470", ¿cómo podría realizar tal operación, ya que poseo los programas "VU-3D" y "PAINTBOX" y por mucho que he intentado hacerlo todavía no he conseguido ni un solo copiado de pantalla.

d) ¿Tiene mi impresora capacidad gráfica suficiente para copiar cualquier clase de dibujo o gráfico que salga por pantalla aunque sean de alta resolución como los que aparecen en el programa "DEMO" del "PAINTBOX"?

Perdonad la extensión de mi carta, pero he decidido dirigirme a vosotros porque creo que si hay alguien que pueda resolverme los interrogantes que planteo con seriedad, concisión y profesionalidad es vuestra revista.

Juan Navarro Posadas
Córdoba

R *Muchisimas gracias. Ponemos tu pregunta en manos del Servicio Técnico de Dirac:*

a) "El programa de copy que nosotros comercializamos, es para realizar el volcado de pantalla del Spectrum a la impresora correspondiente, mediante el interface de Indescomp por conexión paralelo Centronics. El programa reside en las posiciones de memoria 64348 al final de memoria física. Utiliza la misma memoria que el interface de Indescomp por lo que no podrá funcionar con programas que utilicen código de máquina en esas direcciones.

b) Llamaremos al programa de copy simple por RANDOMIZE USR 65044 y al doble RANDOMIZE USR 65047.

c) El programa "PAINTBOX" y "VU-3D" permiten guardar las pantallas realizadas como fichero, por lo que se pueden cargar posteriormente como screen $, para volcarlas a impresora.

d) Sí, perfectamente se pueden realizar en tamaño reducido y ampliado.

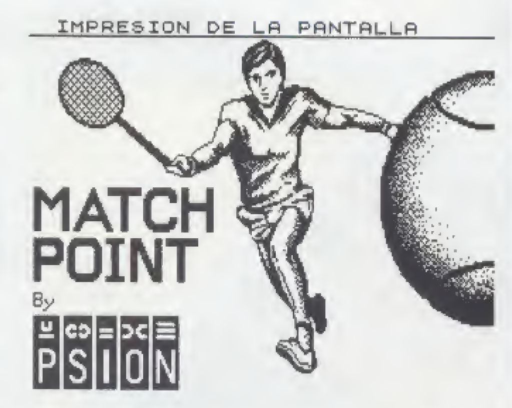
Disponemos de un interface con ROM externa intercambiable llamado MASTERFASE 4, que no ocupa memoria RAM y que realiza el COPY mediante los comandos COPY 1 y COPY 2."

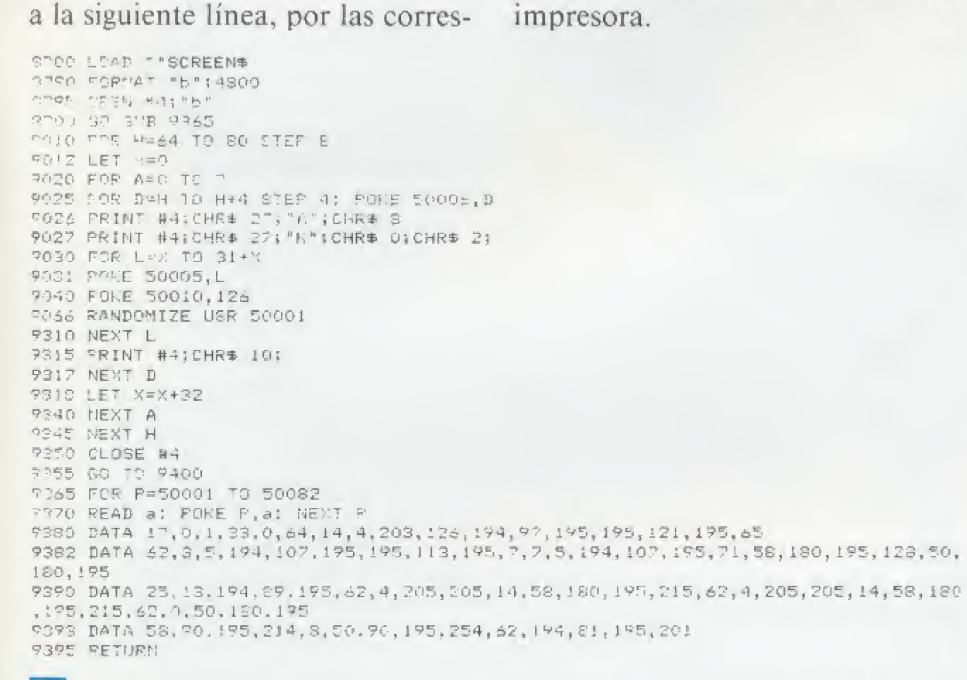
P Como respuesta a la pregunta que os hacía Miguel Angel de

Oviedo en el número de marzo referente a poder copiar la pantalla en la impresora os envío un programa que realiza esta labor.

Mi impresora es una CPA 80 y supongo que las instrucciones de control de impresión son iguales para todas. De no ser así baterías con cambiar las líneas 9026 = espacio entre líneas; que ha de ser de 8 puntos (o 8 bits, es decir, un byte en vertical), 9027 = número de bytes a mandar por línea; que tienen que ser 512 por línea, 9315 = salto a la siguiente línea, por las correspondientes instrucciones de cada impresora.

IMPRESION DE LA PANTALLA





```
9000 LOAD ""SCREEN$
9090 FORMAT "b";4800
9095 OPEN #4;"b"
9005 GO SUB 9365
9010 FOR H=64 TO 80 STEP 8
9012 LET X=0
9020 FOR A=0 TO 7
9025 FOR D=H TO H+4 STEP 4: POKE 50006,D
9026 PRINT #4;CHR$ 27;"A";CHR$ 8
9027 PRINT #4;CHR$ 27;"K";CHR$ 0;CHR$ 2;
9030 FOR L=X TO 31+X
9031 POKE 50005,L
9040 POKE 50010,126
9056 RANDOMIZE USR 50001
9310 NEXT L
9315 PRINT #4;CHR$ 10;
9317 NEXT D
9318 LET X=X+32
9340 NEXT A
9345 NEXT H
9350 CLOSE #4
9355 GO TO 9400
9365 FOR P=50001 TO 50082
9370 READ a: POKE P,a: NEXT P
9380 DATA 17,0,1,33,0,64,14,4,203,126,194,97,195,195,121,195,65
9382 DATA 62,3,5,194,107,195,195,113,195,7,7,5,194,107,195,71,58,180,195,128,50,
180,195
9390 DATA 25,13,194,89,195,62,4,205,205,14,58,180,195,215,62,4,205,205,14,58,180
,195,215,62,0,50,180,195
9393 DATA 58,90,195,214,8,50,90,195,254,62,194,81,195,201
9395 RETURN
```

R *Efectivamente, la copia corresponde al Match Point, que analizaremos próximamente en la sección de juegos.*

P Hace más de dos meses les solicité una copia en cassette del programa de evaluación escolar. Toda vez que no tengo noticias, les ruego contestación por si no he procedido a su petición en la forma adecuada.

Joaquín Rodríguez
Toledo

R *Su petición era "correctísima". La demora ha sido motivada por problemas de fabricación. Confiamos en que todos puedan recibirla en este mes.*

P Quisiera saber si existe alguna otra foma de acceder a los programas contenidos en la cinta de Spectrum Computing que ustedes distribuyen. De no ser así, si a partir de ahora continuarán publicando programas en que sea necesaria la desconexión del interface.

Jordi Millán
Barcelona

R *Efectivamente, diversos programas requieren de la desconexión del interface por utilizar rutinas en código máquina posicionados en la zona de memoria utilizada por aquél. El utilizar estas direcciones de memoria no es un capricho; se trata de utilizar al máximo la capacidad del ordenador y de sacar «bytes» de todos los sitios posibles. En cualquier caso, siempre se puede desconectar el interface.*

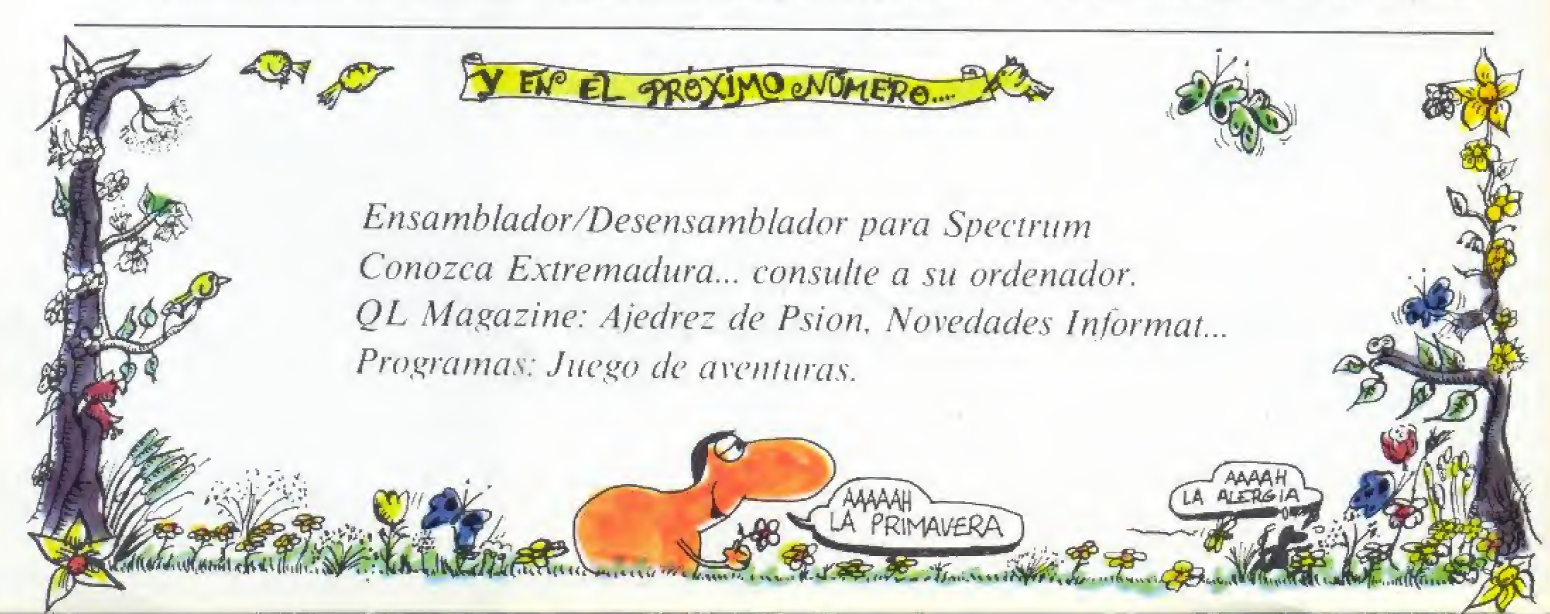


PROGRAMMER II

La *TI Programmer II* de **Texas Instrument** es, como sugiere su nombre, una calculadora diseñada específicamente para programadores y estudiantes de informática en general, que necesiten una herramienta de trabajo rápida y fácil de manejar para realizar diferentes operaciones y cálculos tanto en base 10 (decimal), base 16 (hexadecimal) y base 8 (octal).

La *TI Programmer II* se caracteriza por ser una máquina cómoda y de diseño sencillo, plana y de color gris metálico. Tiene 40 teclas en tres colores diferentes entre los que destacan (además de las operaciones aritméticas usuales [+, –, ×, /], teclas de números del 1 al 9 y 6 teclas para las cifras hexadecimales de la A a la F), las teclas de conversión de una base a otra (DEC, HEX, OCT); teclas de funciones específicas como son: la tecla K (que permite almacenar el número y su operación asociada), la tecla STO (almacena de forma relativa los resultados obtenidos) la tecla RCL (restituye la información almacenada en la memoria), SUM (acumula los resultados de las operaciones realizadas), y otras específicas para utilizar en base Hexadecimal y Octal como AND (Y lógico), OR (O lógico), XOR (O exclusivo), 1'sC (permite la complementación a uno), etc., que pueden ser aplicadas a una serie de funciones aritméticas y lógicas como son: la realización de operaciones aritméticas en las tres bases; complementación a dos en base octal y hexadecimal; trabajar con coma flotante en las operaciones aritméticas en base decimal; conversión de números enteros de una base a otra; 15 niveles de operaciones con paréntesis disponibles; memoria constante e independiente con capacidad para acumular el resultado; funciones lógicas que permiten efectuar las operaciones lógicas bit a bit en hexadecimal y octal; capacidad de trabajar con las operaciones lógicas o aritméticas en modo constante y, por último, puede cortar automáticamente la alimentación de la calculadora después de pasados 20 minutos de no ser utilizada.

Estas características hace de éste modelo de **Texas**, una herramienta potente y eficaz (su consumo no es excesivo a pesar de estar conectada permanentemente) para solucionar los problemas más usuales que se le presentan al programador o estudiante cuando está trabajando, aunque en ciertos aspectos, la máquina pueda estar limitada en determinadas funciones y capacidades como son la realización de operaciones y cálculos más complejos y conversión efectiva al sistema binario. Precio 14.990 ptas.





Periférico del año en UK.

¡Imagínate! Cinco utilísimos componentes integrados en una sola unidad compacta que se conecta a tu Spectrum en unos segundos. Sin cableados engorrosos. Potente y eficaz. Eso es **WAFADRIVE, elegido periférico del año en el Reino Unido y fabricado por Rotronics.** *

Interface con el Spectrum, interface serie RS/232 (con velocidad de transmisión seleccionable por software), interface paralelo Centronics y dos drives de **128 K cada uno** que utilizan cartuchos de 16, 64 o 128 K diseñados para proporcionar una **alta velocidad** de transferencia de datos (2 K por segundo) y la **máxima fiabilidad**.

Dale a tu Spectrum la potencia y la versatilidad de un sistema auténticamente profesional. Y aprovéchate de nuestra **Oferta Especial** de lanzamiento en la que, para que conozcas uno de los muchos programas ya disponibles para el **WAFADRIVE,** incluimos el **Spectral Writer** (un excelente Procesador de Textos). Y un cartucho virgen. Y manuales en castellano, claro.

Todo por 48.500 Ptas. en tu tienda de Informática o directamente en **MICROBYTE**.

* También fabricante de los moduladores ASTEC

Sí, quiero aprovecharme de su Oferta Especial de lanzamiento. Envíenme un WAFADRIVE, el Procesador de Textos Spectral Writer y un cartucho virgen, todo por 48.500.– Ptas.

Nombre

Dirección

Población Código Postal

Teléfono

Incluyo Talón Nominativo ☐

Contra Reembolso ☐

ZX

MICROBYTE, P.º Castellana, 179-1.º
28046-Madrid
Pedidos por teléfono:
91-442 54 33